Chapter 3

# Wireless Sensor Networks: Technology Overview

Wireless Sensor Networks (WSNs) started as a wild academic idea that turned into a very commercially relevant technology. This class of wireless networks has received significant attention in the last decade because of the unprecedented operational conditions it offers. From a number of proof-of-concept demonstrations, WSNs have evolved into a highly reliable, commercialized technology.

From solely sensing the environment, WSNs have evolved to become increasingly integrated with the Internet. Complete standards-based communication stacks are appearing, which enable tiny wireless devices to reliably form a communication mesh, with protocols running on this mesh enabling end-to-end IP connectivity. WSNs are the enabling technology that will help shape what tomorrow's Internet of Things (IoT) looks like.

## 3.1. History and context

The history of WSNs is fascinating: from extremely smart scientists solving head-banging problems, to marketing people

Chapter written by Thomas WATTEYNE and Kristofer S.J. PISTER.

*The Internet of Things: Connecting Objects to the Web* Edited by Hakima Chaouchi

showing once more that (deceptively) simple solutions are the ones that have the most commercial potential.

### 3.1.1. *From smart dust to smart plants*

The smart dust concept, a Defense Advanced Research Projects Agency project funded in 1997, started from the desire to make micro-robots using micro electro-mechanical systems (MEMS) technology. In 1992, it was clear that three different technologies were following exponential curves down to zero cost: sensing (driven by the MEMS revolution); computation (following Moore's law); and communication.

Similarly, it was clear at the time that the size and power of such devices would follow similar trends to cost: everything you needed to build a wireless sensor node was decreasing in size, power and cost. That was the seed of the smart dust idea.

The concept of smart dust resonated with a whole community of people. In 2001, at Intel's development forum, 800 Berkeley motes were placed in the main auditorium, one under each seat. Note that the term "mote", defined as "a speck of dust", has become synonymous with wireless sensors. During the keynote session the next morning, participants were asked to take out the motes. The motes formed a multi-hop communication infrastructure, showing up in real-time on the main screen. Self-healing was demonstrated by pulling out the batteries of randomly chosen motes, and seeing that the network reorganized around the remaining motes. This was the birth of multi-hop, self-organizing and self-healing wireless networking.

In another milestone demonstration in 2001, wireless sensors were placed under the wings of an unmanned aerial vehicle, which was programmed to drop sensors along a road. Once deployed, the motes, equipped with magnetometers, recorded the time of passage of vehicles. The plane flew back and forth along the road, queried each sensor on its passage and reported the vehicle passage times back to a base station.

Commercial analysts in 2003, seeing the success of these academic demonstrations, started seeing commercial potential in this technology. The reason for this continuing enthusiasm is that WSNs, a technology where low-cost sensors can be put anywhere and start reporting data without having to put wires in, can be used almost everywhere. The application space covers fields as diverse as building automation (security, heating-ventilation-air-conditioning (HVAC), Automated Meter Reading (AMR), lighting control, access control), industrial monitoring (asset management, process control, environment and energy management), body sensor networks (patient monitoring, fitness), home electronics (TV, VCR, DVD/CD, game console), computer interfacing (mouse, keyboard, joystick), energy applications (lawn and garden irrigation, energy monitoring, demand-response systems, smart grid), etc.

As a response, the Institute of Electrical and Electronics Engineers (IEEE) started looking at the problem, and in 2003 came up with the first version of its IEEE 802.15.4 standard. This standard defines both the physical and medium access communication layers. The ZigBee industrial consortium came together to build a set of standards on top of IEEE 802.15.4; they produced their first set of drafts in 2004. This solidified interest, and venture capitalists started putting money into the field.

### 3.1.2. *Application requirements in modern WSNs*

A WSN is capable of hitting a hot spot in return-on-investment that a wired sensor network is unable to attain. While Moore's law predicts how the hardware gets cheaper, the installation cost and especially the wiring involved with installing a wired sensor network is sometimes prohibitive.

Think for example of an oil refinery with multiple tanks interconnected by miles of tubing, running a complex industrial process. There will be thousands of sensors monitoring pressure, temperature, flow rate, tank level, valve health, etc. all ready to be hooked up to a central monitoring station. While the more important ones are indeed wired in, the vast majority of those available sensors

are not because of the prohibitive cost of wiring. Using a WSN removes most of the installation overheads: the network can be deployed in hours rather than weeks and it self-organizes to provide the central monitoring station with real-time data on a much larger set of sensors present in the plant.

The real challenge faced by a modern WSN is to provide wired-like reliability using wire-free technologies.

3.1.2.1. *Number, geometry and topology*

The early vision of smart dust led people to think that it would be sprinkled throughout an environment more or less randomly. Some deployments did this, but for the vast majority of sensor network applications today the sensors are individually installed where they are needed.

Some systems are installed by trained technicians and others by doctoral students, both groups that can be counted on to have some sophistication and ability. But the majority of the networks are installed by people who may have no technical background whatsoever.

Today, most sensor networks are not connected to the Internet. Access points are generally plugged into a system that uses the data locally, and the information flows in the network do not extend beyond the sensor network itself. This is likely to change dramatically over the next decade, during which IP-based sensor networking is likely to take off. Many sensor networks will still not connect to the Internet, however, due to tradition, politics or concern over security.

Motes report sensed data from their environment; it hence makes little sense to not know where the reported sensor is located. Think of a warehouse where thousands of items are stored and moved around by forklifts. Imagine now that each of these items is equipped with motes capable of determining their location inside the warehouse. Not only would it no longer be possible to lose items, but a warehouse supervisor could issue a query directly into his or her warehouse find out exactly how many items it contains, and where each is located.

Like in GPS, localization systems use some form of triangulation, where a node measures its distance to a set of location-aware reference nodes. The cornerstone problem with localization is that ranging, i.e. measuring the distance between two nodes, is a non-trivial problem. Techniques such as received signal strength perform badly, especially indoors. In recent years, a technique called radio frequency (RF) time-of-flight has been shown to outperform previous techniques. The idea is to measure the time it takes for a RF packet to travel from the sender to the receiver and back. While this is commonly used in ultra-wide band (UWB) systems, applying this technique to an IEEE 802.15.4 radio (with only 2 MHz-wide channels) is challenging. Interested readers are referred to [LAN 09].

#### 3.1.2.2. *Data flows*

There is such a wide range of applications of sensor networks that virtually any type of data flow can be ascribed to some type of network, real or conjectured. Here, we describe the most common examples, and follow the notation from [RPL 10]. In most networks, there is at least one "special node", that we will call the sink node, which connects to some other information system.

Most reporting in sensor networks is periodic. The period may vary from milliseconds to days, but the hot spot for current technology ranges from seconds to minutes. Events may trigger the flow of data in a multipoint-to-point (MP2P) flow, as in a home alarm system where a door or window opening causes a packet to be sent to the alarm control box. Fault conditions on a mote, or evidence of a security attack, may also generate packets to be sent to the sink. Some systems use reports by exception, where the data are sampled on a regular period, but are only reported if they fall outside of some specified range.

By far the most common data flow in existing sensor networks is the regular collection of data from many points to one collection point, or MP2P. This is such a common flow that we will assume it to be a baseline in all of our discussions, and point out its absence in those rare cases where it does not appear.

In pharmaceutical monitoring, for example, temperature data from dozens or hundreds of sensors is sent back to the data logger attached to the sink node. Network, mote health and status information is often sent to the sink, either to enable network control in a centrally-managed network or for diagnostic purposes in a distributed management system.

Broadcast commands from the sink to some or all of the motes in the network is used for over-the-air programming, changing network parameters such as ID and data-link-layer keys, and synchronous sampling or actuation commands. This traffic flow is called point-to-multipoint.

Finally, point-to-point traffic between motes occurs in control applications. A light switch sending a packet to a light fixture is an example of open-loop control. A tank level sensor sending a packet to a valve is an example of closed-loop control. Most of these flows are short geographically.

3.1.2.3. *Latency-bounded reliability*

The sole purpose of the *networking* piece of WSNs is to deliver data. It is the reliability of that delivery on each of the data flows that sets most of the requirements on the network. Reliability is the fraction of packets introduced to the network that successfully get to their destination. For some applications, a reliability of 90% may be acceptable. For others, the probability that even a single packet is lost out of millions sent must be a tiny fraction of a per cent. Usually, if someone tells you that reliability is not important to them, then there is probably an opportunity to redefine the data flow in a more mote-amenable way. If 50% reliability is acceptable on a flow of one packet per second, then the application would probably be just as happy with one packet every two seconds with 99.9% reliability.

For most data flows, reliability is tied directly to latency. Most applications will not tolerate a network that delivers 100% of the packets after a one-year delay. Some applications will be sensitive to the mean latency, and others will be more concerned with worst-case latency. For example, people are willing to tolerate the occasional

long response time as long as the average is reasonably low, whereas a feedback control system may not care about the mean as long as the worst-case latency is bounded.

3.1.2.4. *Lifetime, cost and size*

With WSNs there are no wires, so energy is a scarce commodity. For a given topology, flow, radio and protocol, the lifetime of a mote is related to the amount of energy it can store or scavenge. Storage and scavenging require both cost and size. In most applications, cost is the driver rather than size.

For example, if C-cell batteries were free, most sensor network applications would use them even though they are somewhat ungainly. In general, the reason that people want small batteries is because they are cheaper. Given the choice between equal cost C-cell and coin-cell batteries, the decision is likely to be made first based on lifetime, and then finally size. If the coin cell only lasts the required lifetime in 80% of the desired deployments, then the larger C-cell is likely to be used. Only when the lifetime and cost are both satisfied is size likely to be the deciding factor.

Clearly, there are exceptions to this. If car batteries were free, they would be too big for most applications. For medical sensors worn on the body, a C-cell is going to be unacceptable for almost every application.

3.1.2.5. *Security*

Most people do not have the nefarious disposition necessary to truly appreciate the need for security. Security people tend to think in terms of the worst-case scenario, and how to exploit weakness and improbable events.

Technologists and entrepreneurs very naturally tend to think about the benefits of their technologies. Embrace that, and imagine that your application is wildly successful, and that people are using it in ways that go beyond what even you initially thought. Sadly, even foolish people are using technologies in ways that they probably should not be used.

Now try to think like a crook, a hacker, a terrorist. Imagine that you have a lot of resources behind you, and try to come up with a set of worst-case scenarios.

## 3.2. The node

A wireless mote contains a small number of integrated circuits, or "chips", connected together onto a circuit board and powered by a battery. The heart of a mote is its micro-controller: a tiny processor into which all the other chips connect. The micro-controller typically coordinates the sampling of the sensor chips and the communication through the radio chip. The radio chip sends the packets it receives from the micro-controller to an antenna. The sensor chips come in many sizes, packages and types, and deliver sensed data either through digital (as a series of 0s and 1s) or analog (as a voltage) ports.

### 3.2.1. *Communication*

Typical transmission power (the power actually radiated out the antenna as RF energy) is in the 1 to 10 mW range. The radio that generates this transmission power can be thought of as having two parts: the modulator, which converts bits into the appropriate time varying (RF) voltage, and the power amplifier that boosts the signal and delivers it to the antenna.

In low-power radios, the modulator often burns more power than the power amplifier. With the power amplifier off, essentially no power goes out of the antenna, so there is a minimum "overhead" of current from the power supply just to generate the appropriate voltages. It is difficult to design a power amplifier that is efficient over a wide range of output powers, and power amplifiers are generally designed to be most efficient near their maximum power output capability. The efficiency of the power amplifier is typically between 10% and 80%, and the lower end of that range is most common for low-power chips.

The result is that a 10 times reduction in the output power of the radio is rarely coupled with a corresponding reduction in radio current. Although most sensor network radios do have some type of transmission power control, often over two orders of magnitude or more, the difference in radio current is rarely greater than two times.

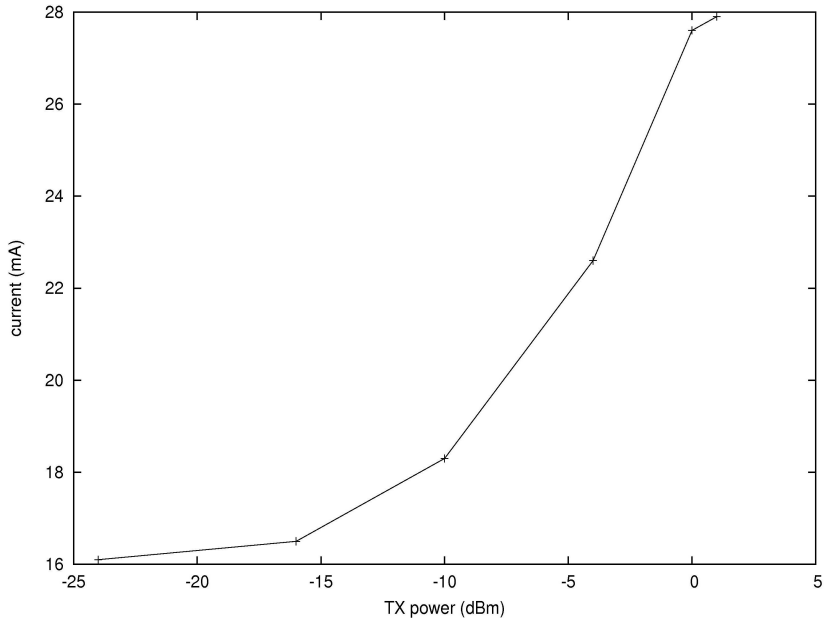


**Figure 3.1.** *Transmitter output power versus input current*

As an example, take the measurements presented in Figure 3.1, obtained from an eZ430-RF2500 board equipped with a CC2500 radio. Note how a roughly 300 times drop in transmission power (from 1 dBm = 1.25 mW to -24 dBm = 0.004 mW) translates into just a 42% decrease of the radio's current consumption (from 27.9 mA to 16.1 mA).

Radios can only receive information if the received signal is strong enough. The minimum detectable signal level for a radio is called its sensitivity. Typical numbers for mote radios are a fraction of a

picoWatt (sensitivities from -90 dBm to -100 dBm are typical). The ratio of transmission power to receiver sensitivity is called the link margin. For a transmitter putting out a few milliWatts, and a receiver with a sensitivity of a few tenths of a picoWatt, the link margin is around 10 billion! In practice, it can sometimes be challenging to receive a message from a mote that is only a few meters away.

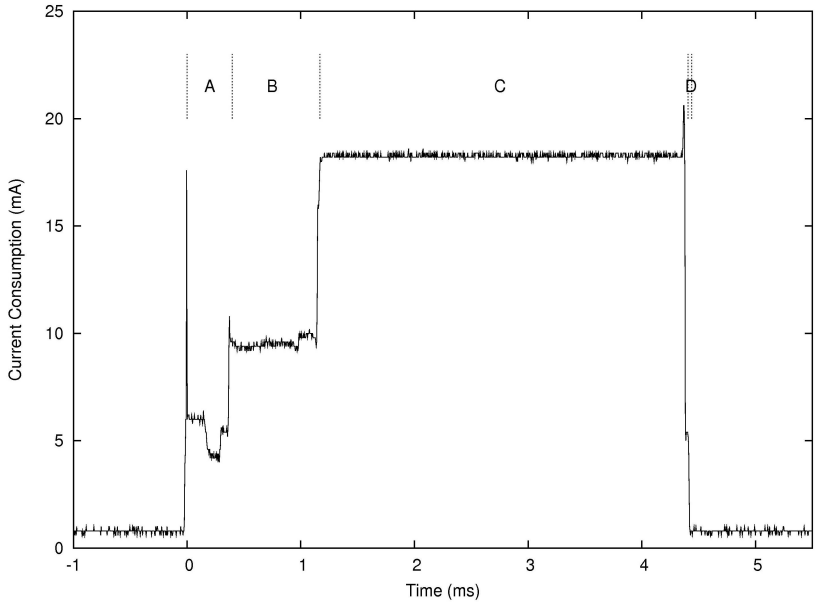


**Figure 3.2.** *Radio current during startup and TX. Results averaged over 128 samples*

Before the first bit can be sent or received on the radio, a long sequence of events must typically take place inside the radio. From a deep sleep state, this entails turning on a voltage regulator, waiting for a crystal oscillator to stabilize, and waiting for the radio oscillator to settle (tune) to the proper frequency, among other things. Figure 3.2 shows how a CC2500 radio takes roughly 1 ms to switch between OFF and RX modes. The annotated phases are radio startup (A), radio frequency calibration (B), reception mode (C) and entering sleep (D).

### 3.2.2. *Computation*

Low power microprocessors typically operate with 8-, 16-, or 32-bit quantities of data. Usually, the instruction width is the same as the data width, although there is now a family of 32-bit processors from ARM that use a 16-bit instruction set. In general, the wider the datapath and instruction, the more you can do in a single instruction and a single clock cycle. So a 20 MHz 8-bit processor will be a lot slower than a 20 MHz 32-bit processor (sometimes more than 10 times), and the code size for the 8-bit processor will be larger than for the 32-bit processor (maybe several dozen percent).

Programs are typically stored in flash memory or ROM. Both of these are non-volatile, meaning that they do not go away when the power is turned off. ROM and flash are both very high density (roughly 1 Mb/mm$^2$ in 0.18 μm complementary metal-oxide-semiconductor, CMOS). Variable data are stored in SRAM (static random access memory), which is about ten times lower density than flash or ROM, so typically there is a lot less of it on a microprocessor. SRAM is volatile, so when the rest of the chip goes to sleep, you still have to keep your SRAM powered if you want to retain any of the information on it. Fortunately, it does not burn much power when it is just sitting there retaining information, since it is just the leakage through all of the transistors. A few μA is typical.

The majority of mote processors spend most of their time sleeping. The software wakes up periodically due to timer interruptions, e.g. sample sensors and send or receive packets, and when this is done the software goes back to sleep. Using a processor that efficiently moves between sleeping and waking states is important for low-power operation.

### 3.2.3. *Sensing*

There are thousands of different kinds of sensors, each with its own interface specifications. Even a single company, selling a single type of sensor, may have dozens of different versions of that sensor

with different performance, interfaces, packaging, temperature tolerance, etc.

There is no such thing as a general sensor interface. Increasingly, sensors have integrated electronics, so they may present either a digital interface or a low impedance voltage output, both of which are relatively easy to interface to a microcontroller.

### 3.2.4. *Energy*

The battery anode and cathode chemistry determine its fresh (pre-discharge) open-circuit voltage as well as the temperature range for normal operation. Lithium batteries have a flat discharge profile, meaning that their voltage remains constant over most of their useful life. Alkaline batteries have a linear decrease in voltage as their capacity is drained. Lithium batteries generally have longer shelf-life due to lower internal leakage. Consumers pay much higher prices for lithium batteries than for alkaline. Lithium thionyl chloride batteries are the most expensive of all, and have the highest performance.

In addition to the current that you pull out of the battery, it will also self-discharge. This limits the lifetime for low current levels. At high currents, the internal resistance of the battery, among other things, reduces the amount of charge available to the application. The useful capacity of a battery is a strong function of temperature, average current and current profile.

## 3.3. Connecting nodes

### 3.3.1. *Radio basics*

If a radio transmits a continuous tone, then there is no information available to the receiver other than the frequency of the transmission. To communicate information, the transmitter must change some aspect of the transmitted wave. This is called modulation. The simplest form of modulation is to turn the tone on and off, which is known as on-off keying, or OOK. Often the tone is not turned all the way off, but rather different amplitudes are used. This is known as

amplitude shift keying, or ASK. If the transmitter modulates the frequency of the radiated wave instead of its amplitude, this is called frequency modulation. This simplest form of frequency modulation is frequency shift keying, or FSK.

Broadcast radio in the so-called FM band (88 – 107 MHz) uses this method. In phase shift keying, or PSK, the signal is transmitted by modulating the phase of the carrier. In quadrature PSK (QPSK), the phase moves between {0, 90, 180, 270} degrees. This corresponds to the transmitting sine waves (0 or 180 degrees) and cosine waves (90 and 270 degrees). Most of the energy in the transmitted wave lies in a frequency band equal to twice the sum of the frequency deviation and the frequency of the modulated data.

The signal coming off the antenna contains many other components in addition to the received power from the transmitter. Most of these components are from undesired radio transmissions, which we will call interference. Some of these are from other man-made sources, such as electric power distribution and electric sparks in rotating equipment such as electric motors and spark plugs. Some noise is from natural sources, such as lightning and solar flares.

The sensitivity of a radio is the minimum received signal power required in order to achieve a specified bit error rate or packet error rate. The required signal power depends on the amount of noise present, and on the minimum signal-to-noise ratio (SNR), needed by the analog-to-digital converter and digital electronics.

The minimum SNR depends on the specified error rate, the modulation used, the algorithm used for demodulation and decoding, and the quality of the implementation of that algorithm. The link margin is a measure of how much power can be lost between the transmitter and the receiver. For typical WSN radios, the transmission power is between 1 and 10 mW, and the sensitivity is between -90 and -100 dBm, giving link margins of between 90 and 110 dB.

### 3.3.2. *Common misconceptions*

A common misconception about wireless communication is that the communication area of a node is a perfect disk of radius $R$. According to this model, all nodes closer than $R$ can hear the node perfectly; nodes further away than $R$ cannot hear it at all. This might be true in the theoretical case of an infinite free space where radios with a perfectly deterministic transmission power and sensitivity communicate using perfectly isotropic antennas.

In reality, no such claim can be made, mainly because of RF phenomena, such as external interference and multi-path fading. These phenomena, which have a greater presence indoors, are detailed in the next sections.

Note that these observations have a profound impact on protocol design. It is, for example, not possible to design a protocol using geometric assumptions; this is, unfortunately, the case in much geographic routing protocol. It is not possible to deterministically tune the communication range of a node either; many poorly designed protocols rely for example on the capability of some node to transmit "twice as far" as others.

A second common misconception relates to energy. A simple rule of thumb is that a radio that is on consumes almost the same amount of energy whether it is transmitting, receiving or idly listening. An energy-efficient protocol should hence maximize the time the radio is turned off rather than, for example, reduce the number of transmitted packets.

A third common misconception is related to ranging capabilities using received signal strength. Most radios, upon receiving a packet, indicate at what power that packet was received. It is tempting to try to relate this power to the distance of the transceiver. For the reasons stated earlier (multi-path, indeterminism in the radio), this assumption does not hold.

### 3.3.3. *Reliable communication in practice: channel hopping*

WSNs face the challenge of ensuring reliable communication over inherently unreliable links. The bad news is that external interference and multi-path fading cause the quality of wireless links to change dramatically, in an unpredictable way. The good news is that these phenomena change depending on the frequency the nodes are communicating on. Channel hopping is a technique proven to efficiently combat the unreliable nature of wireless technology.

Let us take a real-world example. Connectivity traces were collected by Ortiz and Culler in a University College Berkeley office space (connectivity traces are available at http://wsn.eecs.berkeley.edu/connectivity/). 46 IEEE802.15.4-compliant TelosB motes are deployed in a 50 m by 50 m indoor environment and are constantly listening for packets. One after the other, each mote transmits a burst of 100 packets, with a 20 ms inter-packet time and a transmission power of 0 dBm, on each of the 16 frequency channels that span the 2.4-2.485 GHz band. Timers are used to ensure that all nodes switch channels simultaneously. Note that, because bursts are sent in sequence, there are no collisions. All non-transmitting nodes record the time stamp of the packets received, their source address, and the frequency channel the packets are received on. After all 46 nodes have sent a burst, each node reports which packets it has received. This process is repeated in 17 runs. A single run completes in 13 minutes. Several hours separate subsequent runs.

With these traces in hand, we can plot the reliability of a link depending on its frequency. Reliability can be simply expressed as the packet delivery ratio (PDR): the ratio between the number of received packets and the number of sent packets. A PDR of one indicates a perfect link. Figure 3.3 plots the average reliability of all links, depending on their frequency. While at some frequencies (e.g. channel 26, or 2.480 GHz) PDR is around 87%, it drops to close to 75% at others (e.g. channel 12, or 2.415 GHz).

It turns out that the people working on that office space connect to the Internet wirelessly using IEEE802.11 (WiFi) base-stations operating on IEEE802.11 channels 1, 6 and 11. When plotting the

frequency used by those channels in Figure 3.3, it becomes clear that external WiFi interference severely impacts the reliability of the WSN. Does this mean that we should tune the WSN to operate only on, for example channel 20? What if a network administrator then installs a fourth WiFi network operating at the same frequency? Clearly, static channel allocation is not the answer.

In an indoor environment, every wall, person and piece of furniture acts as a reflector for RF signals. As a result, on top of the signal following the direct line-of-sight (LOS) path, a node receives multiple echoes that have bounced off nearby elements. The paths those echoes follow are necessarily longer than the LOS path so they arrive a bit later, typically within a few nanoseconds. This is an unwanted phenomenon, particularly in narrowband communication. If the different signals are phased appropriately, they can destructively interfere and the receiver is unable to decode the signal even when physically close to the transmitter.

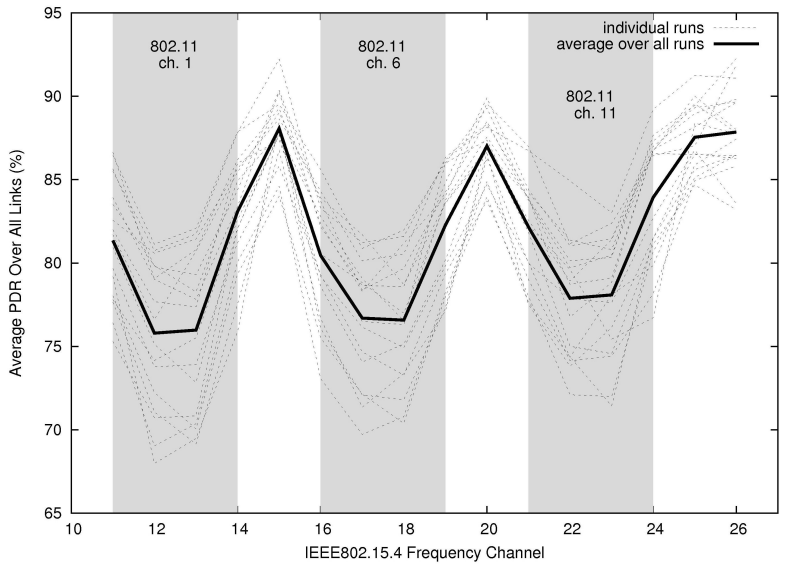


**Figure 3.3.** *Radio current during startup and TX. Results averaged over 128 samples*

Let us take results collected from a real-world experiment taken from [WAT 09]. A computer is connected to a fixed receiver mote; a transmitter mote is mounted on a motorized arm. At the beginning of a measurement, the arm is moved to a given location. The transmitter then transmits 1,000 29-byte-long packets at a given requested frequency. The PDR is determined by the receiver as the fraction of packets that were successfully received. Each of the 1,000 packets take 2.3 ms to be sent; one measurement (including the movement of the arm) takes 4 s. This measurement is repeated for different transmitter locations inside a 20 cm by 34 cm plane; with a 1 cm step in both directions, i.e. 735 data points are acquired.

Figure 3.4 depicts the resulting 3D plot of PDR *versus* transmitter location, when transmitter and receiver are separated by (only) 1 m. This dramatic figure denotes some bad news. While in most locations connectivity is good with PDR hovering around 100% (remember that transmitter and receiver are only 1 m apart, so this result is expected), in some locations PDR drops all the way down to 0%. Even worse, it does so after the transmitter has been moved by only 2-3 cm.

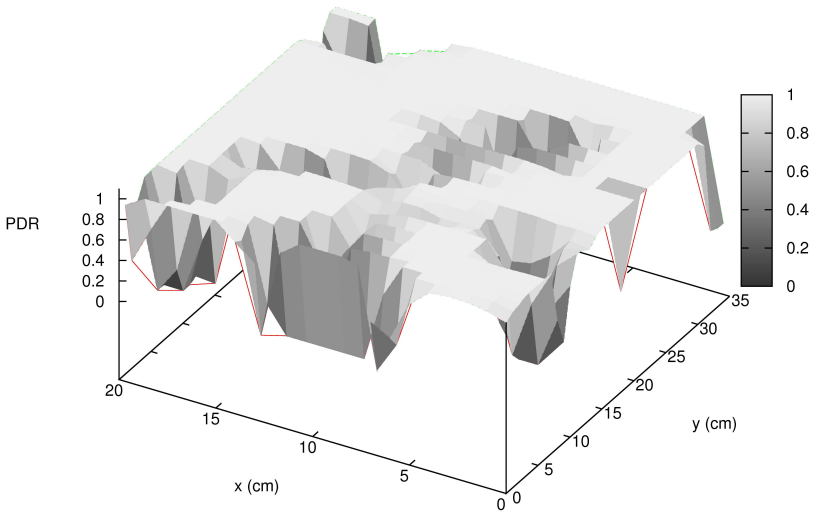


**Figure 3.4.** *Witnessing multipath fading. Results obtained for sender and receiver communicating on IEEE802.15.4 channel 20 (2.450 GHz) separated by 1 m*

For a network designer, this is indeed bad news. Multi-path fading depends entirely on the environment, so it cannot be predicted without infinite knowledge of the object's location, orientation and reflective characteristics. When adding the fact that people walk around, and doors are opened and closed, predicting the location of the deep fades (the location where PDR reaches 0%) is unfeasible.

Yet this phenomenon depends on frequency. Repeating the same measurement for different frequency channels does indeed show that the "topography" of Figure 3.4 changes significantly from one channel to another. In fact, for a transmitter and receiver separated by a couple of meters or more, the impact of the operating frequency is such that a frequency shift of only 5 MHz (one channel in the IEEE802.15.4 standard) leads to an entirely different topography.

So what does that entail for a communication system? The answer is that channel hopping should be used. In a channel hopping system subsequent packets are sent at a different frequency, following a pseudo-random hopping pattern. This means that, if a transmission fails, retransmission will happen on a different frequency. This means that the transmission has a greater chance of being successful than if the retransmission happened on the same channel because a different frequency means different effects of multi-path fading and interference.

## 3.4. Networking nodes

Many things need to happen inside a node for it to be able to communicate over a multi-hop network. The software running on the node needs to answer many questions. At what time should a node send a message? On what frequency? If a node wants to report a measurement to a distant node, to which neighbor should it send its message? What should it do when a transmission fails? Re-transmit? Discard the packet? Change its destination?

A network engineer implements a specific program that runs on the mote. Although it is software, it is typically called "firmware" because it is not supposed to be installed by the end user. Much like when you

buy a washing machine, it comes pre-loaded with some firmware that reacts to you pressing buttons on the machine.

Unlike a washing machine, however, the code running on a communicating device is pretty complex as it needs to take care of many different things. Acknowledging this complexity, in 1977 the International Organization for Standardization defined a generic way of describing any communication system. This model, called the seven-layer open system interconnection reference model, has proven to be generic enough that most communication systems follow it.

Communication layers is the key concept of a communication system; everyone working on networking wireless sensors should have an excellent understanding of related concepts, such as encapsulation, layer interchangeability or interfaces. If you are unfamiliar with the concepts, please refer to [TAN 02].

### 3.4.1. *Medium access control*

The medium access control (MAC) layer, because it deals with two key constraints, is arguably pivotal in WSN communication architecture [LAN 05, DEM 06]. First, it controls the state of the radio chip, hence the duty cycle and the energy-efficiency of the node. Second, since the wireless medium is broadcast in nature, it is in charge of resolving any contention arising, while taking link outages and changes of topologies due to nodes (dis)appearing into account.

There has hence been a growing interest in understanding and optimizing WSN MAC protocols in recent years. Research was driven primarily to reduce energy consumption because of the limited and constrained resources on-board a wireless mote.

All energy-efficient MAC protocols switch the radio off to save energy, while switching it on every now and then to communicate. Different approaches have been taken, which we classify into two big families: preamble sampling protocols and frame-based scheduled protocols [BAC 09].

3.4.1.1. *Preamble sampling protocols*

Nodes using preamble-sampling periodically listen for a very short time (called the clear-channel-assessment, or CCA) to decide whether a transmission is ongoing. The check interval (CI) is the amount of time a node waits between two successive CCAs. The sender needs to make sure the receiver node is awake before sending data; it thus prepends a (long) preamble to its data. By having a preamble at least as long as the wake-up period, the sender is certain the receiver will hear it and be awake to receive the data. Note that this technique has been referred to in literature as cycle receiver [LIN 04], low-power listening [POL 04], channel polling [YE 06], and preamble sampling.

Figures 3.5 to 3.7 are chronographs depicting the radio state of node *S* and its three neighbors A, B and C, for different preamble-sampling variants. A box above/under a vertical line means the node's radio is transmitting/receiving, respectively. No box means the radio is off. All nodes sample the channel for $D_{cca}$ seconds every *CI* seconds.

Figure 3.5 depicts basic preamble-sampling functions: node *S* sends a continuous preamble of length $CI + D_{cca}$. When nodes A, B and C sample the channel, they stay awake until the data message of length $D_{data}$ is sent.

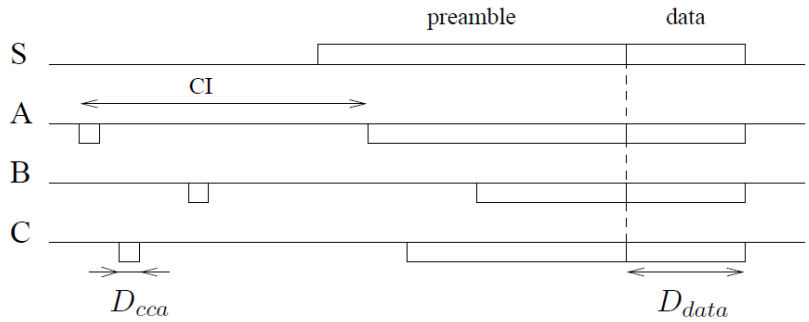


**Figure 3.5.** *Basic preamble-sampling*

Basic preamble-sampling requires A, B and C to listen to the remainder of the preamble, which costs energy. Micro-frame preambling (MFP) [BAC 06] cuts the preamble into a series of micro-frames (see Figure 3.6). Each micro-frame contains a counter indicating how many micro-frames still remain. A micro-frame is sent every $T_{mf}$ seconds, and lasts for $D_{mf}$. Upon sampling the channel, a node knows how many micro-frames are still to be sent, and it can hence return to sleep until the actual data are sent.

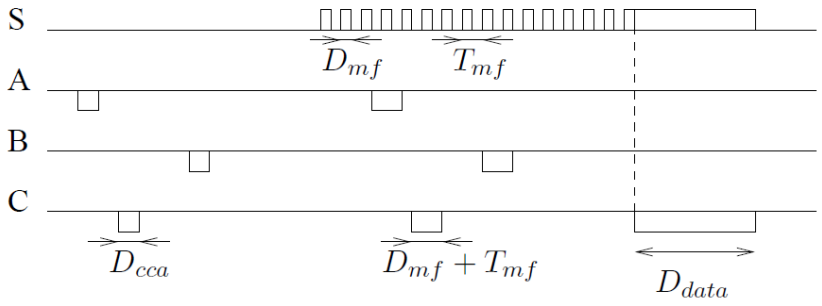


**Figure 3.6.** *Micro-frame preamble-sampling*

One major drawback of preamble-sampling is that preambles are long, which costs energy and increases collision probability. Techniques have been proposed to overcome this problem.

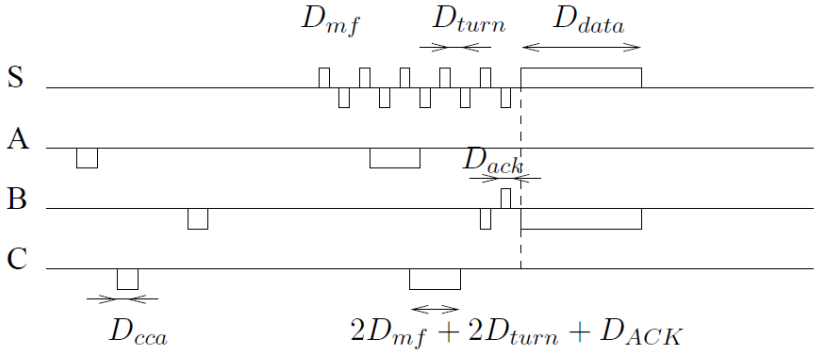


**Figure 3.7.** *X-MAC, MFP with listening between micro-frames*

X-MAC [BUE 06] uses a concept similar to MFP. The sender *S* cuts the preamble into micro-frames, and listens between each micro-frame (see Figure 3.7). Note that *S* needs a time $T_{turn}$ to switch between reception and transmission modes. When the destination node (here B) hears the preamble, it answers with an acknowledgment message of length $D_{ack}$. This causes the length of the preamble to be, on average, half that of MFP.

There is an optimal value for the CI beyond which nodes waste more energy in transmission than they save in reception. Finding this optimal value depends mainly on the traffic load on the network. As an example, let us consider 10 nodes that are all within communication range, and sample the channel for 200 µs every CI. Without traffic, the average duty cycle is $(200.10^{-6})$/CI, so the larger the CI, the more energy efficient the protocol is. Assuming that *a* messages are sent between the 10 nodes per second, we can easily calculate the duty cycle depicted in Figure 3.8 for several loads. The larger the load, the smaller the duty cycle should be.

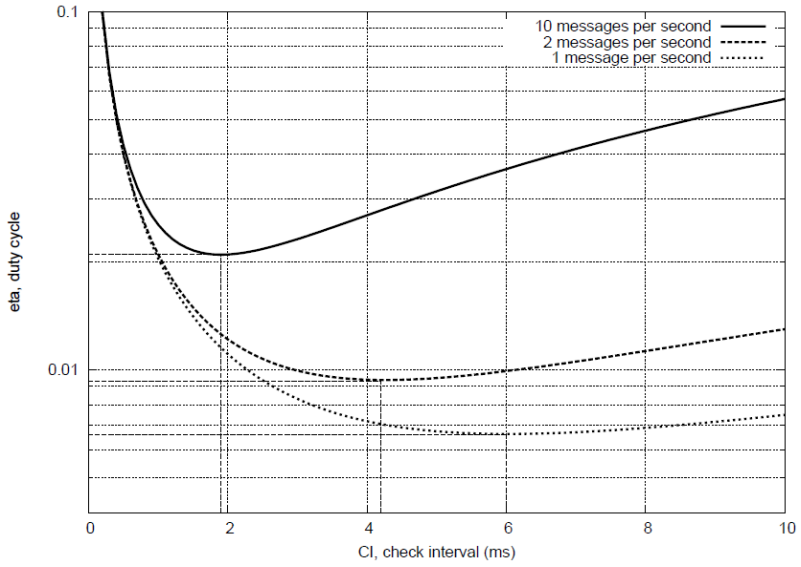


**Figure 3.8.** *In preamble sampling, the optimal check-interval depends largely on the network load*

Preamble-sampling removes the need for control traffic. It offers an elegant "always-on" abstraction that somewhat simplifies the interaction with upper layers, as cross-layer communication is not needed. Preamble sampling, however, suffers from two main problems.

The first is that, by appending preambles, packets get longer and are hence more prone to collisions. This means that preamble-sampling needs to be used for low-throughput networks only. A WSN is most congested at the sink node, as all traffic converges to that network. The rule of thumb is that preamble sampling is only efficient when a sink node receives less than one packet per second. In cases when the traffic increases above that threshold (think of a forest fire detection WSN: where a fire starts, all nodes start generating alarm messages), the network is said to "collapse". When there are many packets, the probability of collision is high, and hence most packets need to be retransmitted, which adds even more to the burden. This causes very few packets to successfully reach the sink, although nodes in the network are constantly trying to communicate. An experimental example of this phenomenon is presented in Figure 3.9.

The second problem is that frequency agility is hard to couple with preamble sampling. Frequency agility is the capability of the network to communicate on different channels, in order to combat external interference and multi-path fading. As we will see in the next section, framed MAC protocols are better able to cope with these problems.

#### 3.4.1.2. *Framed MAC protocols*

Framed MAC protocols construct a schedule that all nodes follow. They are also referred to as time division multiple access (TDMA) protocols. A schedule is a succession of slots that form a frame that continuously repeats over time. Note that slots can be generalized to partition the available resource (channel) along the time, frequency of code axis, or any combination thereof. When a node needs to send a message, it waits for the slot during which it knows no other node is transmitting. This approach is attractive because once the schedule is set up it can significantly reduce the number of collisions, the amount of idle listening and overhearing. This approach offers bounded

latency, fairness and good throughput in loaded traffic conditions, at the cost of reducing adaptability to variable traffic.

Data can be scheduled in different ways into slots. When communication links are scheduled, specific sender-receiver pairs are assigned to a given slot. This avoids both overhearing and collisions, but may decrease network throughput if traffic is variable. Slots can be assigned to senders: during its slot, a node is given the opportunity to transmit to any of its neighbors, requiring all of its neighbors to listen. The opposite is also possible (i.e. scheduling receivers), in which case multiple nodes may end up competing to send, requiring contention-resolution techniques.

Gateway MAC [BRO 06] elects a node acting as a gateway for a certain time, and then rotates nodes in order to balance load. The TDMA frame of gateway MAC contains three periods: the collection period, the traffic indication period and the distribution period.

During the collection period, nodes compete for the channel and send packets expressing their future traffic needs. In the traffic indication period, all nodes wake up and listen to the channel to receive the gateway traffic indication message. The gateway traffic indication message maintains synchronization among nodes and assigns slots to nodes.

The traffic-adaptive medium access protocol [RAJ 03] determines a collision-free scheduling and performs link assignment according to the expected traffic. The protocol contains two phases: localized topology formation and scheduled channel access. The scheduled channel access allows each node to wake up only to transmit or to receive, which reduces idle listening and overhearing to zero. The main issue with traffic-adaptive medium access protocol is its complexity and the assumption that nodes are synchronized network-wide.

Y-MAC [KIM 08] is primarily designed to decrease latency. Nodes are synchronized and reception slots are assigned to each node on a common base channel. In cases where multiple packets need to be sent between neighboring nodes, successive packets are sent, each on a

different frequency following a pre-determined hopping sequence. This hopping sequence starts at the base channel. As a result, bursts of messages ripple across channels, significantly reducing latency. The implementation results presented serve as proof-of-concept for the multi-channel MAC approach.

Time-synchronized mesh protocol (TSMP) [PIS 08] is TDMA-based and hence requires network-wide synchronization. Access is controlled by means of a tunable amount of time slots that form a frame. The protocol is designed so that a node can participate in multiple frames at once, allowing it to have multiple refresh rates for different tasks. In addition, TSMP employs frequency division multiple access (FDMA) and frequency hopping. Different links use differing frequency slots and the same link hops during their lifetime across different frequency slots. This yields high robustness against narrow-band interference and other channel impairments.

A traditional approach to facilitate synchronization is beaconing. Longer frames decrease synchronization refresh rate and power consumption; shorter frames invoke the opposite. TSMP nodes maintain a sense of time by exchanging resynchronization messages during active periods together with the usual data and acknowledgment packets; this invokes negligible overheads. TSMP nodes are active in three states: 1) sending a packet to a neighbor; 2) listening for a neighbor to talk; and 3) interfacing with an embedded hardware component.

The duration of active periods, i.e. duty cycling, is very flexible in TDMA; typical applications require duty cycles of less than 1% on average.

When applied, the sink typically retrieves the list of nodes, their neighbors and their requirements in terms of traffic generation. From this information, it constructs a scheduling table in both time and frequency. When implementing TSMP on IEEE802.15.4-2006 [802.15.4] hardware, 16 frequency channels are available. Exemplified by the scheduling table in Figure 3.9, the TSMP link establishment and maintenance rules are simple: never put two transmissions in the same time/frequency slot; at a given time, a given

node should not receive from two neighbors or have to send to two neighbors. Assuming that slots are 10 ms long and node H sends a packet following route H – F – B – G, then H send to F in slot [t5, ch.6], thereafter F – B in [t10, ch.11], then B – G at [t8, ch.8]. Latency can in this case be reduced to three slots or 30 ms. Figure 3.9 shows that successive packets traveling between two nodes are sent using different frequencies, following a preset hopping sequence.

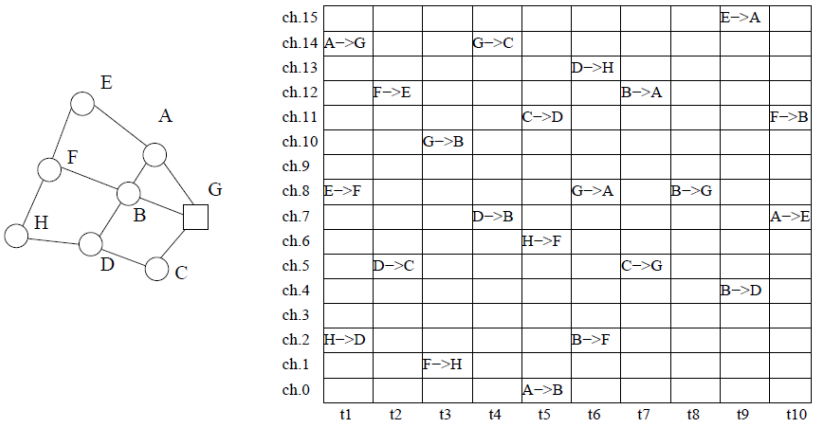


**Figure 3.9.** *An (naive) example of a TSMP scheduling table for the graph depicted on the left*

TSMP uses multi-channels not to increase network throughput, but to increase robustness against narrow-band interference. Figure 3.9 shows that successive packets traveling between two nodes are sent using different frequencies, following a preset hopping sequence. [DOH 07] presents experimental results in which 44 nodes were deployed running TSMP, including retransmission mechanisms, for 28 days in a printing facility. A delivery ratio of over 99.999% was reported.

Figure 3.10 shows the superiority of framed MAC protocols at high loads. Results were obtained experimentally from our TinyOS 2.1 implementation on TelosB motes (complete source code is available at http://wsn.eecs.berkeley.edu/). We call the number of packets successfully received per second at the sink “goodput”. The

idle duty cycle is the portion of time a node has its radio active when there is no traffic on the network. In the experimental setting, a node running TSMP has an idle duty cycle of 2%. For fair comparison, we set the check interval of preamble sampling to 58 ms, which yields the same idle duty cycle. Note that results for preamble sampling are worse at lower idle duty cycles.

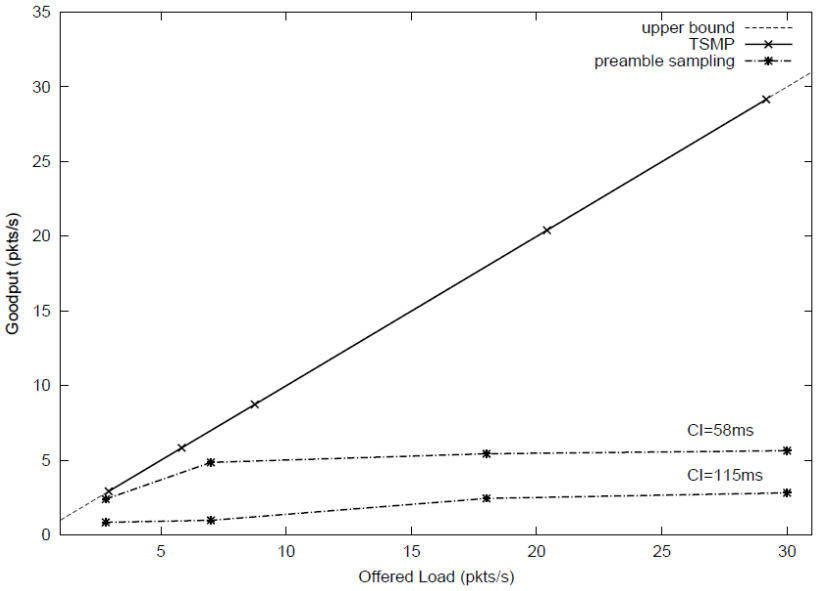


**Figure 3.10.** *Comparing the goodput of TSMP and preamble sampling shows the greater value of the former at high loads*

We use a simple star topology, where a sink node has seven neighbors constantly sending data to it. As shown in Figure 3.10, when we crank up the load of the network, a preamble sampling MAC protocol quickly plateaus at around three to five packets per second; TSMP, in contrast, is capable of supporting loads above 30 packets per second. More generally, TSMP achieves significantly higher throughput, lower energy consumption and smaller average single-hop latency than preamble sampling, over a wide range of offered loads.

### 3.4.2. *Multi-hop routing*

In large networks, a data source may not reach the intended sink in a single hop, thereby requiring the traffic to be routed via multiple hops. An optimized choice of such a routing path is known to significantly increase the performance of said network. There has hence been a growing interest in the understanding and optimizing of WSN routing and networking protocols in recent years. The limited and constrained resources here have driven research towards reducing energy consumption, memory requirements and complexity of routing functionalities.

To this end, early flooding-based and hierarchical protocols have migrated within the past decade to geographic and self-organizing coordinate-based routing solutions. The former have been inspired by MANET (Mobile Ad-hoc NETwork) -type approaches; the latter are currently finding their way into standardization.

Thanks to the work of several generations of researchers working on this problem, different approaches have contributed to a now solid body of knowledge. The field has reached a state of maturity that enables standardization organizations to aggregate several elements from that body into a standard. One of these standardization organizations is the Internet Engineering Task Force (IETF), ubiquitous in today's Internet protocols.

Research on a multi-hop wireless network has continued through a few eras. Initially, these networks were envisioned to be constituted of highly mobile nodes (e.g. cars, handhelds, etc.) wanting to exchange large amounts of data without real energy concerns. IETF's MANET work group was thus created in 1998. The evolution of the needs and the fact that the MANET charter aimed to solve an incredibly complex problem has led to the initial vision being changed. Most wireless multi-hop networks are now seen as being constituted of highly energy constrained and static wireless sensors transmitting very small quantities of data. In 2008, IETF's routing over low power and lossy networks (ROLL) was created to standardize a routing protocol for such WSNs.

Within the 10 years separating those two visions, network requirements have evolved to a point where solutions for MANET-type networks no longer apply to WSNs. Flooding-based and hierarchical protocols (developed by MANET) are being replaced by geographic and self-organizing coordinate-based routing solutions. IETF ROLL is in the final stages of standardizing a solution based essentially on self-organizing coordinate systems, called RPL (IPv6 Routing Protocol for Low power and Lossy networks) [RPL 10].

3.4.2.1. *IETF MANET: a complex inheritance*

Historically, routing protocols developed for mobile ad hoc networks have been adapted to the new needs of WSNs. These protocols are particularly interesting for coordinating small groups of mobile nodes. They deliver data without the need for any routing algorithms and topology maintenance. This happens at the price of each sensor node broadcasting the data packet to all of its neighbors, with this process continuing until the packet arrives at the destination or the maximum number of hops for the packet is reached. Numerous variants to this protocol have been developed to improve on the energy efficiency. These have been discussed in [LEV 09, ALK 04].

Dynamic source routing (DSR) [JOH 07] performs on-demand route discovery and source routing of packets. It maintains a source route for all destinations. The route to the destination is learned after a discovery phase started by the source that floods route request packets in the network. Each crossed node adds its identifier to the packet and continues forwarding it to all of its neighbors, until the packet reaches the destination node. The destination node then sends a route reply message that follows the inverse path of the request (stored in the packet). DSR does not require sequence numbers or other mechanisms to prevent routing loops because there is no problem of inconsistent routing tables.

Ad hoc on demand vector routing (AODV) [PER 03] is a distance-vector protocol intended for MANETs. AODV is on demand so it only maintains routes for active nodes. As in DSR, when one AODV node requires a route to another node, it floods the network with a request to discover a route. AODV chooses routes that have the minimum hop

count. If a route request packet reaches a node that has a route to the destination (or that is the destination itself), then that node sends a reply along the reverse route. All nodes along the reverse route can cache the route without the need to include the routing state in the packet's header. When routes break due to topology changes, AODV floods error messages and issues a new request.

Dynamic mobile on-demand routing (DYMO) [CHA 08] is an evolution of AODV. The basic functionality is the same, but it has different packet formats and handling rules, and supports path accumulation. Path accumulation allows a single route request to generate routes to all nodes along the path to that destination. Like AODV, DYMO uses hop counts as a routing metric, but it can assign to a link a cost higher than one. Like AODV, on link breaks, DYMO issues a new route request message with a higher sequence number so that nodes do not respond from their route caches but flood the packet into the network.

With reference to the above-discussed constraints, flooding-based routing protocols clearly do not cater for parameter constrained routing as the protocol class at hand requires large energy expenditures, albeit low with memory and little computational complexity. Neither is it optimized for the converge-cast traffic patterns or is it scalable. Furthermore, since no attempt is undertaken to compute the shortest or optimum routing path, latency is clearly an issue. Also, to implement viable security measures using such energy-consuming protocols seems unrealistic. The protocol class, however, adapts very quickly to any link unreliability or network dynamics. Finally, it does not require any form of human intervention and hence facilitates autonomous network operation.

As stressed by [DOL 07], while WSNs and ad hoc networks are both wireless multi-hop networks, they are different in three aspects: 1) energy efficiency is a primary goal for WSNs; 2) in most envisioned applications, the amount of data transported by a WSN is low; and 3) all information flows towards a limited number of destinations in WSNs. Routing protocols designed for ad hoc networks are thus inadequate in large and dense sensor networks [LEV 09].

3.4.2.2. *Geographic routing*

Many WSN applications (e.g. tracking location lions in a national park) require all nodes to know where they are. In outdoor applications, this may be achieved through GPS, but any other method is possible. With the application requiring location-awareness, there is no overhead for reusing this location information for communication purposes. This is the philosophy behind geographic routing, which uses the knowledge of a node's position together with the positions of its neighbors and the sink node to elect the next hop.

Greedy geographic routing is the simplest form of geographic routing [STO 05, FIN 87]. When a node receives a message, it relays the message to the neighbor geographically closest to the sink. Irrespective of the definition of proximity, greedy routing can fail when a node has no neighbor closer than itself to the destination.

More advanced geographic routing protocols guarantee delivery under the assumption of reliable links and nodes. The key idea of these protocols is to switch between two modes. The default mode uses the greedy approach described above. In case this mode fails, a second mode is used to circumnavigate the void area. Once on the other side of this void area, the greedy mode is resumed.

Bose *et al*. propose greedy-face-greedy routing [BOS 99], which uses this principle. Greedy-face-greedy switches from greedy mode to face mode when a void is met. Face mode is used to circumnavigate the void using the right-hand rule. When the current node is closer to the destination than the node initially starting the face mode, the protocol returns to greedy mode.

One important aspect for this to work is that, in face mode, the protocol must only consider the edges between itself and its neighbors that are on the planar subgraph. Planarity is achieved by virtually removing crossing edges from the connectivity graph. Techniques relying solely on geometric considerations, such as Gabriel graph [GAB 69] transformation, suffer from the fact that they assume the communication area of the nodes is a perfect disk, which we have seen does not hold. Techniques that can actually be implemented can

only be made to work at considerable overhead [KIM 05]. Although geographical routing protocols are hardly practical, they have opened the path to gradient-based protocols.

3.4.2.3. *Gradient routing*

The concept of gradient is particularly useful for converge-cast networks, such as WSNs. In the simplest collection scenario, all traffic is sent to a single sink node. In this case, a single gradient – rooted at the sink node – is built and maintained in the network. Figure 3.11 depicts a topology where nodes are assigned heights calculated as a function of hop count. When node Y at height 3 sends a message, Y sends the message to its shortest neighbor height I; similarly node I relays the message to G, and G to A.

Gradient-based routing [SCH 01] is the canonical gradient routing protocol. On top of the basic idea described above, an energy-based scheme can be used as a data dissemination technique, where a node increases its height when its energy drops below a certain threshold so that other sensors are discouraged from sending data to it.

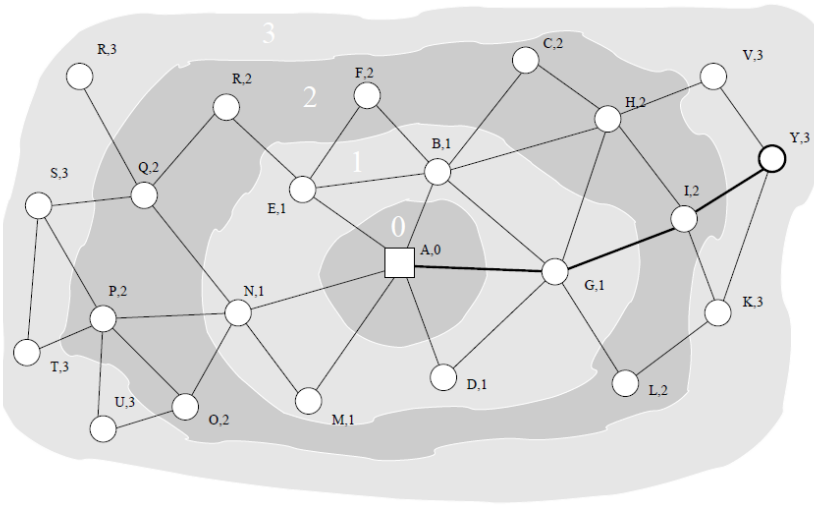


**Figure 3.11.** *Illustrating gradient routing. Nodes are attached [Id,Height]*

Gradient broadcast (GRAB) [YE 05] enhances the reliability of data delivery through path diversity. GRAB builds and maintains a gradient, providing each sensor the direction to forward sensing data. However, unlike all the previous approaches, GRAB forwards data along a band of interleaved mesh from each source to the receiver.

To collect data reports, the sink first builds a gradient by propagating advertisement packets in the network. The height at a node is the minimum energy overhead required to forward a packet from this node to the sink along a path; nodes closer to the sink have a smaller cost. GRAB makes the assumption that each node has the means to estimate the cost of sending data to nearby nodes, e.g. through SNR measurements of neighbors' transmissions. Each node keeps the cost of forwarding packets from itself to the sink. Since only receivers with smaller costs may forward the packet at each hop, the packet is forwarded by successive nodes to follow the decreasing cost direction towards the bottom of the cost field, i.e. the sink.

Multiple paths of decreasing cost can exist and interleave to form a forwarding mesh. To limit the width of this mesh in order to avoid creating excessive redundancy and wasting resources, a source assigns a credit to its generated packet. The credit is extra budget that can be consumed to forward the packet. The sum of the credit and the source's cost is the total budget that can be used to send a packet to the sink along a path. A packet can take any path that requires a cost of less than or equal to the total budget. Multiple nodes in the mesh make collective efforts to deliver data without relying on any specific node.

Performance analysis of GRAB shows the advantage of interleaved mesh over multiple parallel paths and shows that GRAB can successfully deliver over 90% of packets with relatively low energy cost, even under the adverse conditions of node failures and link message losses.

The collection tree protocol (CTP) [GNA 09] uses expected transmission count (ETX) as a link metric for setting up the gradient. Using ETX, the height of a node indicates how many times a message originated at that node is transmitted before it reaches the sink. These

transmissions include the hops from node to node, as well as the retransmissions needed upon link failure.

CTP piggybacks gradient setup information in beacon messages, and uses the trickle algorithm [LEV 04] to regulate the beaconing interval. In the absence of topological changes, this interval is regularly doubled until it reaches a maximum value that triggers only a few beacons per hour. Upon topological changes, the interval is reduced to allow for fast gradient re-convergence. Experimental results on 12 different testbeds show that CTP requires 73% fewer beacons than a solution with a fixed 30-second beacon interval, for an idle duty cycle of 3%.

The IETF, through its ROLL working group, has identified gradient routing as particularly suited for WSNs. It is standardizing the IPv6 routing protocol for low power and lossy networks (RPL, pronounced "ripple") [RPL 10], which captures most of the ideas exploited by the academic proposals listed above. RPL represents, to our knowledge, the state-of-the-art in gradient routing for collection WSNs.

In RPL, a gradient (called directed acyclic graph) is defined by the following four elements: 1) a set of sink node(s); 2) the set of atomic metrics collected on each link (e.g. bandwidth, packet delivery ratio, etc.); 3) how these atomic metrics are combined to obtain the link's cost (by adding, multiplying, etc. the atomic metrics); and 4) how link costs are combined to form a multi-hop path cost (by adding, multiplying, etc. the link costs).

A given network can contain multiple gradients. As an example, depicted in Figure 3.12, consider a building equipped with a WSN in which some nodes (represented by white disks) monitor the power consumption of appliances in the building. These nodes report to a single meter *e* in a way so as to extend the network lifetime. This translates into the following gradient constraints: it is grounded at node *e*, ETX is the link cost, and each node calculates its height as the minimum among its neighbors of that neighbor's ETX, plus the ETX of the link to that neighbor.

Other nodes (represented by shaded disks) are attached to smoke detectors, and report alarms to either one of two fire-monitoring hubs *j* and *k*. Communication between the smoke detectors and the hubs needs to happen with the lowest possible latency. A given network can contain multiple gradients.

As an example, depicted in Figure 3.12, consider a building equipped with a WSN in which some nodes (represented by white disks) monitor the power consumption of appliances in the building. These nodes report to a single meter *e* in a way so as to extend the network lifetime. This translates into the following gradient constraints: the gradient is grounded at node *e*, and ETX is used for the link cost.

Other nodes (represented by shaded disks) are attached to smoke detectors, and report alarms to either one of two fire-monitoring hubs *j* and *k*. Communication between the smoke detectors and the hubs needs to happen with lowest possible latency. This translates into the following gradient constraints: the gradient is grounded at nodes *j* and *k*, and latency is used for the link cost.

In Figure 3.12, latency and ETX metrics are attached to each link; these are used to calculate the latency and ETX heights of each node. When node *a* has to transmit an alarm packet that is intended for either *j* or *k*, it chooses its neighbor with the lowest latency height (here node *c*); by repeating this process at each hop, the packet follows path $a - c - i - j$. Similarly, a packet sent by node *c* follows the ETX gradient, i.e. sequence $c - d - e$.

RPL is strictly compliant with IPv6 architecture; all the signaling used to set up and maintain the gradients are carried as options to the IPv6 packets' router advertisements (RAs). These packets are periodically exchanged between neighbors in the network. To avoid unnecessarily exchanging maintenance traffic while the gradient is stable, the RA period is governed by the trickle algorithm in a fashion similar to CTP [GNA 09].

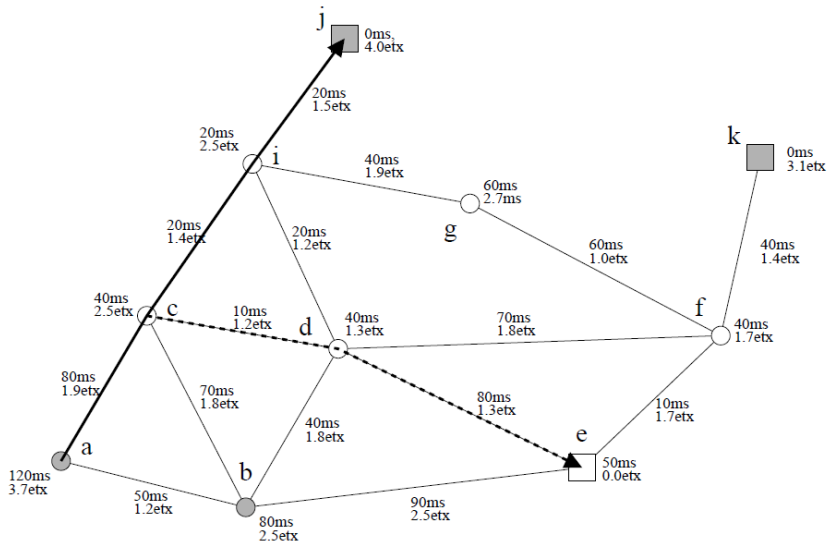


**Figure 3.12.** *A typical building monitoring WSN running IETF ROLL's RPL protocol*

## 3.5. Securing communication

Confidentiality and data integrity in WSNs most commonly uses the same symmetric key encryption technology (advanced encryption standard (AES) cipher in CCM* mode) as commonly found in much larger systems. With built-in hardware support for AES, most motes can perform security operations on an entire packet for less energy than the cost of sending a single bit over the radio.

Authentication, binding and key exchange can be more challenging. Consider the challenge of installing a new wireless light switch into the existing wireless lighting control network at your workplace. Ideally the new device would discover the network, join, and take its role in the network with a bare minimum of human interaction. This is not an easy task. First, your building has a wireless security system, wireless asset tracking and wireless fire alarms, all running similar protocols. Role profiles can help solve this problem, so your light switch knows that it is a light switch, and should not try to talk to the fire extinguisher. Due to the nature of wireless, however,

your switch can hear four different lighting control networks: yours, the ones on the floors above and below yours, and the one across the street. Assuming the switch can pick the right network, it needs to exchange some key information. This cannot be sent in the clear, because the college students have been writing papers about "sniffer-net", and they will take over your building's lights at night and make a lightshow. Finally, once the key material is exchanged, you need a way to tell your new light switch that it is supposed to control the light over your desk, and not your office-mate's.

There are many clever solutions to all of these problems, but so far there are no general solutions that span all application domains. Public key infrastructure is a very powerful tool that is often used in the broader Internet for similar purposes. While public key, or asymmetric key, algorithms are substantially more computationally challenging than symmetric key algorithms, they can still be used on even the smallest wireless sensor platforms [GUR 04].

### 3.6. Standards and Fora

While Chapter 7 covers standardization in great detail, we would like to point out how standardization will play an important role in the future of WSNs as an enabling technology for the IoT. Thanks to the maturity of the field and the unprecedented operating condition WSNs offer, the following standardization bodies have started working on these networks.

The HART Communication Foundation standardizes complete embedded networking solutions for industrial applications. Their wireless extension, called WirelessHART [wHA 08], uses a central controller to schedule communication. It uses IEEE802.15.4 radios to hop on 15 frequency channels in the 2.4 GHz band. Based on TSMP, reliability is increased by having each node maintain connectivity to at least two parent nodes in the routing graph, enabling the network to resist link failures. Additionally, whitelisting is a user-configurable feature of the controller, based on the proximity of other wireless networks that are in the same physical environment.

The International Society of Automation has created a similar standard, ISA100.11a. This standard is also based on TSMP, yet features many different interesting channel-hopping mechanisms. Successive channels in the hopping pattern can be by at least 15 MHz (three IEEE802.15.4 channels). When retransmissions occur, they will not encounter or cause interference in the same IEEE802.11 (Wi-Fi) channel. Moreover, whitelisting limits operation to a subset of channels. At a global scope, a system manager can block certain radio channels that are not working well or are prohibited by local policy. At a local scope, adaptive channel hopping enables whitelisting on a link-by-link basis. The MAC layer of a node bans channels that it deems problematic due to a history of poor connectivity, potentially with granularity of a specific channel used to communicate with a specific neighbor.

Similarly, the IETF has started working on WSNs, standardizing the wire/link and the application. Working groups of greatest interest are:

– IETF ROLL, which standardizes the routing protocol RPL described in the previous section;

– IETF 6LoWPAN, which standardizes the mechanisms for an IPv6 packet to travel over networks of devices communicating using IEEE802.15.4 radios;

– IETF 6LowApp, which studies an application protocol solution for embedded context transfer using appropriate interaction models and a compact binary format compatible with UDP.

Finally, the IEEE, which standardizes the physical layer of the transmitter and the MAC protocol rules, developed the following standards applicable to the IoT:

– IEEE802.15.4 [802.15.4], the technology used by ZigBee, Wireless HART, ISA100.11a, and IETF 6LoWPAN; and by the far the most popular link layer technology that is being adopted for WSNs;

– to a lesser extent, IEEE802.15.1 [802.15.1], the technology used by the Bluetooth consortium;

– to a lesser extent, IEEE802.11 [802.11], the technology used by WiFi.

Note that the next revision of the IEEE802.15.4 standard will redefine the medium access control layer to allow for truly reliable, multi-hop and low-power communication. The solution being finalized, called time synchronized channel hopping, is based on the TSMP MAC layer, hence offering its robustness against external interference and persistent multipath fading. An open-source implementation of TSCH for TelosB motes on TinyOS is available at http://wsn.eecs.berkeley.edu/.

## 3.7. Conclusion

WSNs have witnessed a tremendous upsurge in recent years, both in academia and industry; this is mainly attributed to their unprecedented operating conditions and a variety of commercially viable applications. Such networks can be used in a wide variety of applications, ranging from defense and surveillance to health and intelligent homes.

The real challenge faced by a modern WSN is to provide wired-like reliability using wireless technologies, while remaining low-power to ensure adequate lifetime for these battery-operated devices. Communication protocols have gained sufficiently in maturity that standardization bodies have started working on the field. Standards are an essential step towards large-scale adoption, and with upcoming standards being finalized by mid-2010, WSNs are becoming a key enabling technology that will help the IoT become truly ubiquitous.

## 3.8. Bibliography

[ALK 04] AL-KARAKI J. N. KAMAL A. E., "Routing techniques in wireless sensor networks: A survey", *IEEE Wireless Communications*, vol. 11, no. 6, p. 6–28, 2004.

[BAC 06] BACHIR A., BARTHEL D., HEUSSE M., DUDA A., "Micro-frame preamble MAC for multihop wireless sensor networks", *International Conference on Communications (ICC)*, Istanbul, Turkey, IEEE, 2006.

[BAC 09] BACHIR A., DOHLER M., WATTEYNE T., LEUNG K. K., "MAC essentials for wireless sensor networks", *IEEE Communications Surveys and Tutorials,* forthcoming 2010.

[BOS 99] BOSE P., MORIN P., STOJMENOVIC I., URRUTIA J., "Routing with guaranteed delivery in ad hoc wireless networks", *3rd ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIAL)*, p. 48–55, Seattle, WA, USA, ACM, 1999.

[BRO 06] BROWNFIELD M. I., MEHRJOO K., FAYEZ A. S., DAVIS N. J. I., "Wireless sensor network energy-adaptive MAC protocol", *Consumer Communications and Networking Conference (CCNC)*, p. 778–782, IEEE, 2006.

[BUE 06] BUETTNER M., YEE, GARY V., ANDERSON E., HAN R., "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks". *4th International Conference on Embedded Networked Sensor Systems (SenSys)*, Boulder, Colorado, USA, ACM, 2006.

[CHA 08] CHAKERES I., PERKINS C., "Dynamic MANET on-demand (DYMO) routing", Internet-draft, IETF MANET, draft-ietf-manet-dymo-16., 2008.

[DEM 06] DEMIRKOL, I., ERSOY, C., ALAGOZ, F., "MAC protocols for wireless sensor networks: a survey", *IEEE Communications Magazine*, vol. 6, p. 115–121, 2006.

[DOH 07] DOHERTY L., LINDSAY W., SIMON J., "Channel-specific wireless sensor network path data", *16th International Conference on Computer Communications and Networks (ICCCN)*, p. 89–94, Turtle Bay Resort, Honolulu, Hawaii, USA, IEEE, 2007.

[DOL 07] DOHLER M., BARTHEL D., MARANINCHI F., MOUNIER L., AUBERT S., DUGAS C., BUHRIG A., PAUGNAT F., RENAUDIN M., DUDA A., HEUSSE M., VALOIS F., "The ARESA project: facilitating research, development and commercialization of WSNs", *4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, p. 590–599, San Diego, CA, USA, 2007.

[FIN 87] FINN G. G., "Routing and addressing problems in large metropolitan-scale Internet works", *Technical Report ISI/RR-87-180*, Information Sciences Institute, 1987.

[GAB 69] GABRIEL K., SOKAL R., "A new statistical approach to geographic variation analysis", *Systematic Zoology*, vol. 18, p. 259–278, 1969.

[GNA 09] GNAWALI O., FONSECA R., JAMIESON K., MOSS D., LEVIS P., "Collection tree protocol", *7th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Berkeley, CA, USA, ACM, 2009.

[GUR 04] GURA N., PATEL A., WANDER A., EBERLE H., SHANTZ S.C., "Comparing elliptic curve cryptography and RSA on 8-bit CPUs", *Cryptographic Hardware and Embedded Systems (CHES)*, p. 119-132, 2004.

[802.15.4] IEEE, "Part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs), "*IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – specific Requirements*, IEEE, 2006.

[802.15.1] IEEE, "Part 15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPANs)", *IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan area Networks – Specific Requirements*, 2005.

[802.11] IEEE, "Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications", *IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan area Networks – Specific Requirements*, IEEE, 2007.

[ISA 09] International Society of Automation, *ISA-100.11a-2009: Wireless Systems for Industrial Automation: Process Control and Related Applications*, IHS, 2009.

[JOH 07] JOHNSON D., HU Y., MALTZ D., "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4", *RFC 4728*, IETF, 2007.

[KIM 05] KIM Y.-J., GOVINDAN R., KARP B., SHENKER S., "Geographic routing made practical", *2nd Symposium on Networked Systems Design & Implementation (NSDI)*, p. 217–230, Boston, MA, USA, ACM, 2005.

[KIM 08] KIM Y., SHIN H., CHA H., "Y-MAC: An energy-efficient multichannel MAC protocol for dense wireless sensor networks", *International Conference on Information Processing in Sensor Networks (IPSN)*, p. 53–63, St. Louis, Missouri, USA, IEEE, 2008.

[LAN 05] LANGENDOEN K., HALKES G., "Energy-efficient medium access control", *Embedded Systems Handbook*, CRC press, p. 1–31, 2005.

[LAN 09] LANZISERA S., "RF ranging for location awareness", PhD thesis, University of California, Berkeley, 2009.

[LEV 04] LEVIS P., PATEL N., DAVID C., SHENKER S., "Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks", *Symposium on Networked Systems Design and Implementation (NSDI)*, San Francisco, CA, USA, 2004.

[LEV 09] LEVIS P., TAVAKOLI A., DAWSON-HAGGERTY S., "Overview of existing routing protocols for low power and lossy networks", IETF draft, IETF ROLL. draft-ietf-rollprotocols-survey-07 (work in progress), 2009.

[LIN 04] LIN E.-Y., RABAEY J., WOLISZ A., "Power-efficient rendez-vous schemes for dense wireless sensor networks", *IEEE International Conference on Communications*, vol. 7, p. 3769–3776, Paris, France, 2004.

[PER 03] PERKINS C., BELDING-ROYER E., DAS S., "Ad hoc on-demand distance vector (AODV) routing", *RFC 3561*, IETF, 2003.

[PIS 08] PISTER K., DOHERTY L., "TSMP: Time synchronized mesh protocol", *Parallel and Distributed Computing and Systems (PDCS)*, Orlando, Florida, USA, 2008.

[POL 04] POLASTRE J., HILL J., CULLER D., "Versatile low power media access for wireless sensor networks", *Second ACM Conference on Embedded Networked Sensor Systems (SenSys)*, p. 95–107, Baltimore, MD, USA, 2004.

[RAJ 03] RAJENDRAN V., OBRACZKA K., GARCIA-LUNA-ACEVES J., "Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks", *SenSys*, Los Angeles, CA, USA, ACM, 2003.

[RPL 10] "RPL: IPv6 routing protocol for low power and lossy networks", RPL, forthcoming.

[SCH 01] SCHURGERS C., SRIVASTAVA M. B., "Energy efficient routing in wireless sensor networks", *Military Communications Conference (MILCOM)*, vol.1, p. 357–361, McLean, VA, USA, IEEE, 2001.

[STO 05] STOJMENOVIC I., OLARIU S., "Geographic and energy-aware routing in sensor networks", *Handbook of Sensor Networks: Algorithms and Architectures*, John Wiley & Sons, Inc., Hoboken, New Jersey, p. 381–416, 2005.

[TAN 02] TANENBAUM A. S., *Computer Networks*, 4th edition, Prentice Hall, 2002.

[WAT 09] WATTEYNE T., LANZISERA S., MEHTA A., PISTER K., *Mitigating Multipath Fading through Channel Hopping in Wireless Sensor Networks*, under review, 2009.

[wHA 08] HART, *HART Field Communication Protocol Specifications, Revision 7.1, DDL Specifications*, HART 2008.

[YE 05] YE F., ZHONG G., LU S., ZHANG L., "GRAdient Broadcast: A robust data delivery protocol for large scale sensor networks", *ACM Wireless Networks*, vol. 11, p.285–298, 2005.

[YE 06] YE W., SILVA F., HEIDEMANN J., "Ultra-low duty cycle MAC with scheduled channel polling", *4th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, p. 321–334, Boulder, Colorado, USA, ACM, 2006.