AN ULTRA-LOW DUTY CYCLE SLEEP SCHEDULING PROTOCOL STACK FOR WIRELESS SENSOR NETWORKS

by

Christo Kleu

Submitted in partial fulfilment of the requirements for the degree

Master of Engineering (Computer Engineering)

in the

Faculty of Engineering, Built Environment and Information Technology
Department of Electrical, Electronic and Computer Engineering

UNIVERSITY OF PRETORIA

March 2012

"Research is to see what everybody else has seen, and to think what nobody else has thought."

Albert Szent-Gyorgyi

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

SUMMARY

AN ULTRA-LOW DUTY CYCLE SLEEP SCHEDULING PROTOCOL STACK FOR WIRELESS SENSOR NETWORKS

by

Christo Kleu

Supervisor:	Prof. Gerhard Hancke (University of Pretoria, South Africa)
Department:	Department of Electrical, Electronic and Computer Engineering
University:	University of Pretoria
Degree:	Master of Engineering (Computer Engineering)

Keywords: Wireless sensor networks, energy management, energy efficiency, duty cycle, medium access control, geographical routing, sleep scheduling, clock drift.

A wireless sensor network is a distributed network system consisting of miniature spatially distributed autonomous devices designed for using sensors to sense the environment and cooperatively perform a specific goal. Each sensor node contains a limited power source, a sensor and a radio through which it can communicate with other sensor nodes within its communication radius. Since these sensor nodes may be deployed in inaccessible terrains, it might not be possible to replace their power sources.

The radio transceiver is the hardware component that uses the most power in a sensor node and the optimisation of this element is necessary to reduce the overall energy consumption. In the data link layer there are several major sources of energy waste which should be minimised to achieve greater energy efficiency: idle listening, overhearing, over-emitting, network signalling overhead, and collisions. Sleep scheduling utilises the low-power sleep state of a transceiver and aims to reduce energy wastage caused by idle listening. Idle listening occurs when the radio is on, even though there

is no data to transmit or receive. Collisions are reduced by using medium reservation and carrier sensing; collisions occur when there are simultaneous transmissions from several nodes that are within the interference range of the receiver node. The medium reservation packets include a network allocation vector field which is used for virtual carrier sensing which reduces overhearing. Overhearing occurs when a node receives and decodes packets that are not destined to it. Proper scheduling can avoid energy wastage due to over-emitting; over-emitting occurs when a transmitter node transmits a packet while the receiver node is not ready to receive packets.

A protocol stack is proposed that achieves an ultra-low duty cycle sleep schedule. The protocol stack is aimed at large nodal populations, densely deployed, with periodic sampling applications. It uses the IEEE 802.15.4 Physical Layer (PHY) standard in the 2.4 GHz frequency band. A novel hybrid data-link/network cross-layer solution is proposed using the following features: a global sleep schedule, geographical data gathering tree, Time Division Multiple Access (TDMA) slotted architecture, Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), Clear Channel Assessment (CCA) with a randomised contention window, adaptive listening using a conservative timeout activation mechanism, virtual carrier sensing, clock drift compensation, and error control.

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

OPSOMMING

SLAAP-SKEDULERING VIR DRAADLOSE SENSOR-NETWERKE MET 'N ULTRA-LAE DIENSSIKLUS

deur

Christo Kleu

Studieleier: Prof. Gerhard Hancke (Universiteit van Pretoria, Suid-Afrika)
Departement: Departement van Elektriese, Elektroniese en Rekenaar-Ingenieurswese
Universiteit: Universiteit van Pretoria
Graad: Magister in Ingenieurswese (Rekenaar-Ingenieurswese)

Sleutelwoorde: Draadlose sensor netwerke, energie-beheer, energie-doeltreffendheid, dienssiklus, medium-toegangsbeheer, geografiese data-rapportering, slaap-skedulering, klok-wegdrywing.

'n Draadlose sensor-netwerk is 'n verspreide netwerk stelsel wat bestaan uit miniatuur ruimtelik verspreide outonome toestelle wat ontwerp is om in harmonie saam die omgewing te meet. Elke sensor nodus besit 'n beperkte bron van energie, 'n sensor en 'n radio waardeur dit met ander sensor nodusse binne hulle kommunikasie radius kan kommunikeer. Aangesien hierdie sensor nodusse in ontoeganklike terreine kan ontplooi word, is dit nie moontlik om hulle kragbronne te vervang nie.

Die radio is die hardeware komponent wat van die meeste krag gebruik in 'n sensor nodus en die optimalisering van hierdie element is noodsaaklik vir die verminder die totale energieverbruik. In die data-koppelvlak laag is daar verskeie bronne van energie vermorsing wat minimaliseer moet word: ydele luister, afluistering, oor-uitstraling, oorhoofse netwerk seine, en botsings. Slaap-skedulering maak gebruik van die lae-krag slaap toestand van 'n radio met die doel om energie vermorsing wat veroorsaak word

deur ydele luister, te verminder. Ydele luister vind plaas wanneer die radio aan is selfs al is daar geen data om te stuur of ontvang nie. Botsings word verminder deur medium bespreking en draer deteksie; botsings vind plaas wanneer verskeie nodusse gelyktydig data stuur. Die medium bespreking pakkies sluit 'n netwerk aanwysing vektor veld in wat gebruik word vir virtuele draer deteksie om afluistering te verminder. Afluistering vind plaas wanneer 'n nodus 'n pakkie ontvang en dekodeer maar dit was vir 'n ander nodus bedoel. Behoorlike skedulering kan energie verkwisting as gevolg van oor-uistraling verminder; oor-uistraling gebeur wanneer 'n sender nodus 'n pakkie stuur terwyl die ontvang nog nie gereed is nie.

'n Protokol stapel is voorgestel wat 'n ultra-lae slaap-skedule dienssiklus het. Die protokol is gemik op draadlose sensor-netwerke wat dig ontplooi, groot hoeveelhede nodusse bevat, en met periodiese toetsing toepassings. Dit maak gebruik van die IEEE 802.15.4 Fisiese-Laag standaard in die 2.4 GHz frekwensie band. 'n Nuwe baster data-koppelvlak/netwerk laag oplossing is voorgestel met die volgende kenmerke: globale slaap-skedulering, geografiese data rapportering, Tyd-Verdeling-Veelvuldige-Toegang (TVVT) gegleufde argitektuur, Draer-Deteksie-Veelvuldige-Toegang met Botsing-Vermyding (DDVT/BV), Skoon-Kanaal-Assessering (SKA) met 'n wisselvallige twis-tydperk, aanpasbare slaap-skedulering met 'n konserwatiewe aktiverings meganisme, virtuele draer-deteksie, klok-wegdrywing kompensasie, en fout beheer.

ACKNOWLEDGEMENTS

I would like to extend my appreciation to the following people and institutions:

- Prof Gerhard Hancke, for his continued support and guidance as my supervisor throughout this research.
- Mrs Heleen Gous and Mrs Mari Ferreira, for their help with the administrative tasks.
- Telkom Centre of Excellence (CoE) for granting me a bursary. Without their financial support this research would not have been possible.
- The University of Pretoria for granting me a postgraduate bursary.
- Centre for Teletraffic Engineering in an Information Society (CeTEIS) for their financial assistance for conference attendance.
- Mrs Lindie Kleu, for the language editing of this dissertation.
- A special thanks to my family and my girlfriend, Marike, who supported and motivated me throughout my postgraduate studies.

LIST OF ABBREVIATIONS

ACK	Acknowledgement
ARQ	Automatic Repeat Request
CCA	Clear Channel Assessment
CTS	Clear to Send
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
IEEE	Institute of Electrical and Electronic Engineers
ISM	Industrial, Scientific and Medical
MAC	Medium Access Control
NAV	Network Allocation Vector
PAN	Personal Area Network
PHY	Physical Layer
RF	Radio Frequency
RTS	Request to Send
RX	Receive
SYNC	Synchronisation packet
TDMA	Time Division Multiple Access
TX	Transmit
OSI	Open Systems Interconnection
WSN	Wireless Sensor Network
QoS	Quality of Service

TERMINOLOGY

Clock drift	The phenomena where a clock has a small random deviation from its nominal frequency.
Duty cycle	The time a system spends in an active state as a fraction of the total time under consideration.
Frame	The wireless contention medium is divided into ordered frames and frames are further divided into time slots. The frame length determines the sleep schedule period.
Network convergence	The stable state reached when all nodes within the network have associated with the global sleep schedule.
Node	An autonomous wireless device that forms part of the wireless sensor network.
Packet	The formation of aggregated bits that are transmitted together in time across the physical wireless medium.
Sleep schedule	The schedule which describes the wake-up pattern for a specific node.
Time slot	The time period allocated for a specific level of the data gathering tree structure. The collection of aggregated time slots form a frame.
Data sink	A specialised node that acts as a gateway between the wireless sensor network and the transit network.
SYNC	The periodic synchronisation beacon message which contains the geographical coordinates and sleep schedule information.
Time synchronisation	The process of synchronising all clocks within the network.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

A wireless sensor network (WSN) is a distributed network system consisting of miniature spatially distributed autonomous devices (nodes) designed for using sensors to sense the environment and cooperatively perform a specific goal. Due to the advances in semiconductor technology, network communication, embedded systems and many others, sensor nodes can now be designed and integrated into devices much smaller and cost effective than before. Although the processing power efficiency has increased, the increased density and size of next-generation wireless sensor networks call for even more energy efficient design solutions.

Each sensor node contains a limited power source (battery), a sensor and a radio through which it can communicate with other sensor nodes within its communication radius [1]. Since these sensor nodes may be deployed in inaccessible terrains, it is not possible to replace their power sources. Hence, after a certain amount of time some sensor nodes may die because of a lack of power or some other failure. The focus on energy conservation is extremely important to enhance the lifetime of the network. Most of the attention on wireless standards, i.e. IEEE 802.11 and cellular networks, focus on high rate and long-range applications; the design constraints for most WSNs propose solutions that steer towards low cost, low power and short-range communications.

There are several major sources of energy wastage in the MAC layer of WSNs which should be minimised to achieve greater energy efficiency [2]. The first is *idle listening*, which occurs when the radio transceiver is active while there is no data to transmit or receive. It has been studied that sending and receiving packets of various sizes indicate that the power consumed when the interface is active or idle is virtually identical [3]. The second is *overhearing*, which occurs when a node receives and decodes packets that are not destined to it. The third, *over-emitting*, occurs when the transmitter node transmits a packet while the receiver node is not ready to receive it. The fourth source is the *network signalling overhead*; signalling overhead in WSN protocols can result from per-packet overhead (encapsulation headers and trailers) or from the exchange of control packets. The last major source of energy waste is *collisions*, which occur when there are simultaneous transmissions from several nodes that are within the interference range of each other. Since the collided packets must be retransmitted, it causes unnecessary energy waste.

Researchers have proposed sensor and radio sleeping techniques to address the problem of idle listening [4–11]. Despite the fact that sleeping helps conserve energy, it can have a negative impact on the network performance. For example, a sleeping sensor may result in interesting events being missed by the network or may lead to a lower quality of sensed data. Radio sleeping can cause communication delays in the network since the transmitting sensor nodes need to wait for the receiving sensor nodes to become active. Efficient sleep scheduling is very important since it is a trade-off between the increased network lifetime and quality of timely information that can be gathered.

1.1 RESEARCH QUESTION

Precision agriculture is a farming management concept based on the science of observing, assessing and controlling agricultural practices. A facet of precision agriculture concentrates on site-specific crop management. This encompasses different aspects, such as monitoring soil, crop and climate in a field; generalising the results to a complete parcel; providing a decision support system for delivering insight into possible treatments, field-wide or for specific parts of a field; and the means for taking differential action, for example, varying in real-time an operation such as fertilizer, lime and pesticide application, tillage, or sowing rate.

The research question addressed in this dissertation concerns the utilisation of the low cost and flexibility of WSN deployments in environmental monitoring applications such as precision agriculture. Precision agriculture is described by periodic measurements and a low data rate requirement; in periodic measurement applications, sensors are tasked to periodically sample and report measured values.

1.2 SCOPE

The scope of this research is the development of an energy efficient and adaptive protocol stack for low data rate wireless connectivity with fixed devices and with very limited energy consumption requirements, typically operating in a range of about 10 metres.

The aim is to use various combined techniques to provide efficient sleep scheduling and increasing the network lifetime while monitoring the trade-off between increased network lifetime and latency, bandwidth and reliability.

1.3 MOTIVATION

Wireless sensor networks have recently gained much attention of researchers as they hold the potential to revolutionise many segments of our economy and life: from environmental monitoring and conservation [12–16], manufacturing and business asset management [17], to automation in the transportation and health-care industries [18–20].

The design, implementation, and operation of a sensor network requires the confluence of many disciplines, including signal processing, networking, embedded systems, information management, and distributed systems. These networks are often deployed in resource-constrained environments.

Energy management is a critical issue in sensor networks because of the resource-constrained environments and having to operate without battery recharges for long periods of time while still providing full connectivity [1]. Energy management is required at all layers and components of the sensor network and it impacts the design of all algorithms. Each node must be designed to minimise its local energy consumption while still being able to complete its tasks and minimise global energy consumption without adding too much network overhead.

1.4 OBJECTIVES

The objective of the research is to design an ultra-low duty cycle sleep scheduling protocol stack for wireless sensor networks that satisfies the following requirements.

1.4.1 Optimisation goals

Quality of Service The design should ensure that traffic arrives at the sink in a timely manner.

Energy efficiency The design should strive to extend network lifetime while still adhering to the Quality of Service (QoS) considerations.

Scalability The design must be scalable and function efficiently for networks of any size.

Robustness The design should ensure that the global performance of the system is not sensitive to individual device failures.

1.4.2 Protocol specifications

Latency Data should arrive at the sink before the originating node generates another measurement.

Duty cycle An ultra-low (below 1 %) duty cycle sleep schedule should be achieved.

Self-configuration The design should ensure the ability of network nodes to detect the presence of other nodes and the ability to organise into a structured, functioning network without human intervention.

1.5 RESEARCH METHODOLOGY

A Design Research Methodology procedure, as outlined in Figure 1.1, was followed. The research design was described by a theoretical hypothesis based on previous literature and validation via simulations (using realistic models based on empirical data).

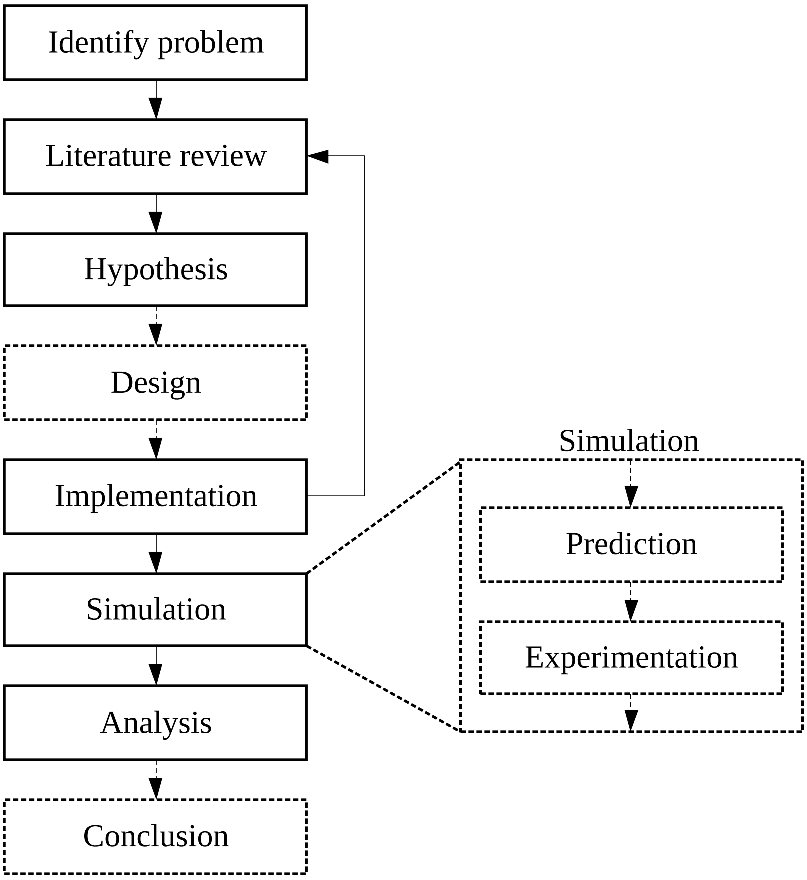


FIGURE 1.1: Research methodology.

Firstly, a suitable topic had to be found by characterising a research problem. The research problem was identified after a broad literature review around WSNs. Secondly, a literature study was performed covering the areas that encompass the research problem. Next, an hypothesis based upon the literature and research problem was formed, and a design was developed to validate said hypothesis.

The implementation stage followed after the design. It should be noted that an iterative process was followed: the initial literature review did not cover certain aspects unique to the implementation process. After the implementation process the design validation experiments were performed. A prediction was made for each experiment, and the simulation experimentation followed.

The final stages included the analysis and conclusion. The analysis involved evaluating the experimental results and a comparison to the simulation predictions. A conclusion was formed by objectively evaluating the simulation results by comparing it to the design specifications.

1.6 CONTRIBUTION

The research contributed to the body of knowledge in the field of WSNs by showcasing a design that proved that it is possible to create a protocol stack that achieves a high level of energy efficiency while still adhering to strict QoS requirements.

The protocol stack is aimed at large nodal populations, densely deployed, with periodic sampling applications. It uses the IEEE 802.15.4 PHY standard in the 2.4 GHz frequency band. A novel hybrid data-link/network cross-layer solution was proposed: a geographically time-slotted architecture using CSMA/CA and an adaptive sleep schedule. Sleep scheduling helps to improve energy efficiency and prolong network lifetime. The design compensated for clock drift errors that occur in oscillator crystals.

1.7 OUTLINE

The dissertation has been structured as follows: Chapter 2, Chapter 3, and Chapter 4 represent the literature study; Chapter 5, Chapter 6, and Chapter 7 represent the design and simulation; Chapter 8 concludes the dissertation.

Chapter 2 presents an overview of WSNs. Chapter 3 provides a literature study of sleep scheduling techniques. Chapter 4 explores the layers of a WSN communication protocol stack. The protocol design is given in Chapter 5. The simulation experiments are described in Chapter 6. The results and discussions can be found in Chapter 7. The dissertation is concluded with Chapter 8 which provides a summary of the proposed design and presents future work and challenges to the research community.

CHAPTER 2

WIRELESS SENSOR NETWORK OVERVIEW

Objectives of this Chapter

This chapter introduces the concept of Wireless Sensor Networks (WSNs) and provides a high-level overview. At the end of this chapter the reader should have an understanding of the design considerations, hardware components, and characteristics of WSNs. A survey of IEEE 802.15 Standards and open-source network simulators are included for reference.

A WSN is a distributed network system consisting of miniature spatially distributed autonomous devices (nodes) designed for using sensors to sense the environment and cooperatively perform a specific goal. Due to the advances in semiconductor technology, network communication, embedded systems and many others, sensor nodes can now be designed and integrated into devices much smaller and cost effective than before. Although the processing power efficiency has increased, the increased density and size of next-generation WSNs call for even more energy efficient design solutions.

One of the most important design considerations of WSNs is to reduce energy consumption. Hence, there is a need for energy-efficient communication and routing techniques that will increase the network lifetime. However, due to limited computing and storage resources the sensors are not provided with an enriched operating system that can provide energy efficient resource management. Application developers are responsible for incorporating energy-efficient communication and routing strategies for WSN applications.

2.1 HARDWARE COMPONENTS

As with all design decisions for wireless sensor nodes, the hardware selection process is also subject to the design considerations (discussed in Section 2.3). Application requirements such as cost, size, and energy efficiency play a vital role in component selection. Figure 2.1 shows the key components that make up a typical WSN node [21].

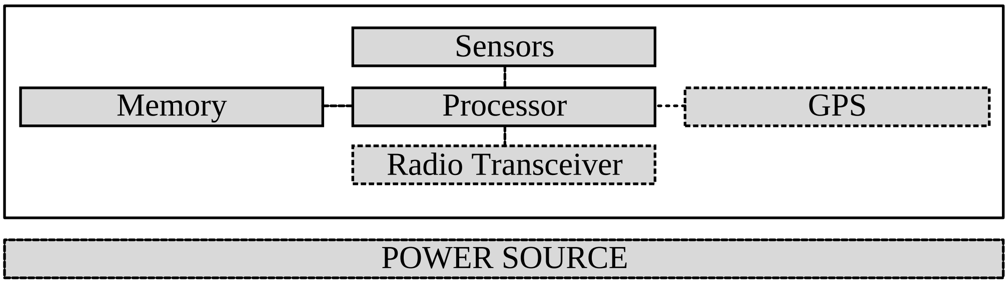


FIGURE 2.1: Schematic of a basic wireless sensor network device.

A basic sensor node includes five main components: an embedded processor, memory, radio transceiver, sensor(s), and a power source. A sensor node can include additional components such as geopositioning systems, mobilisers, actuators, and energy harvesting components.

Low-power embedded processor The computational tasks of a WSN node include the processing of locally sensed information and routed information received by other nodes. The computational power is very limited. Specialised embedded operating systems have been designed for resource-constrained devices, such as TinyOS [22]/Contiki [23].

Memory/storage Device storage includes read-only memory for the operating system and programme storage, and random-access memory for the programme running-configuration and sensor measurements. Complex protocols need more memory. The low data rate of WSN transceivers require additional memory for the buffering of data.

Radio transceiver The radio transceiver allows for low-rate, short-range, wireless communication. The radio provides the essential wireless communication capabilities, but is the most power-intensive operation in a WSN node. Effective techniques are required to reduce the energy consumption.

Sensors Multiple sensors can be used by a sensor node. Sensors are determined by the application and may consist of many different types: seismic, magnetic, thermal, visual, infrared, acoustic, or radar.

Power source Most deployments include battery operated WSN nodes. The finite battery energy provides a limited network lifetime; most WSN protocols aim to address the issues surrounding this resource bottleneck. In the case of heterogeneous deployments, certain nodes could have additional power or energy harvesting abilities.

Geopositioning system Optional: Satellite-based geopositioning systems can be used to provide accurate positioning for localisation and as a source for time synchronisation.

2.2 APPLICATIONS

A WSN generally observe systems that are too complex to be simulated by computer models based directly on their physics. Sensor networks may consist of many different types of sensors such as seismic, magnetic, thermal, visual, infrared, acoustic or radar. These sensors are able to monitor a wide variety of ambient conditions that include, but are not limited to: temperature, humidity, vehicular movement, lightning conditions, pressure, soil makeup, noise levels, object tracking and identification, mechanical stress levels on attached objects and current characteristics such as speed, direction, and size of an object [24, 25].

There are many proposed and implemented applications for WSNs. The design challenges for each of the applications are different, which call for a vast range of different WSN infrastructures, protocols and architectures. Table 2.1 provides a list of projects that use distributed information gathering using WSNs. Other application examples include [2]: disaster relief applications, facility management, machine surveillance and preventive maintenance, precision agriculture and smart transportation.

TABLE 2.1: Examples of prototyped/proposed applications for WSNs.

Type	Project	Description
Environmental monitoring	Great Duck Island [12]	Habitat monitoring the Leach's Storm Petrel breeding patterns.
	ALERT [13]	Flood protection using rainfall sensors, water level sensors, and weather sensors.
	Vineyard Computing [14]	Proposal for using the potential of WSNs in precision agriculture.
	PODS [15]	Ecological monitoring of threatened/endangered plant species and the intensive monitoring of the weather in the area in which the plants occur.
	CORIE [16]	Environmental observation and forecasting system for the Columbia River.
Building automation	Intelligent buildings [18]	The controlling, monitoring, measurement and communication of HVAC parameters (Heating, Ventilation and Air conditioning) in an industrial setting.
Health	Artificial Retina [19]	Biomedical chronically implanted artificial retina with sufficient visual functionality to allow persons without vision or with limited vision to "see" at an acceptable level.
	Body Area Networks [20]	Proposed patient monitoring system: temperature, blood pressure, heart rate, ECG, EEG, respiration rate, SpO_2-levels, etc.
Logistics	Fleet monitoring [17]	Proposed real time tracking and monitoring of conditions in whole supply chain (fresh food fleet monitoring), and providing statistics and market feedback analysis based on the collected data.

2.3 DESIGN CONSIDERATIONS

The architecture of WSNs draws upon many disciplines, including signal processing, networking, embedded systems, information management, and distributed systems. The unique requirements of WSNs have brought forth many different protocols to fill the gap where the existing technologies do not adequately address the design challenges. Protocol design is an engineering decision making process: weighing the trade-off options for the application design considerations.

There are many different types of applications, but the following categories have been identified based on interaction patterns between sources and sinks: event detection, periodic measurements, function approximation and edge detection, and tracking. In this dissertation the focus is on periodic measurement applications; in periodic measurement applications sensors are tasked to periodically sample and report measured values.

The optimisation goals and key design factors are discussed in Section 2.3.1 and Section 2.3.2 respectively.

2.3.1 Optimisation goals

A key challenge in design for WSNs is to find the correct optimisation solution for a given application. Applications have different goals and therefore require different optimisation solutions. A large variety of possible applications have categorized some optimisation aspects.

2.3.1.1 Quality of Service

Depending on the type of application, data transfer can have different performance attributes. Latency and bandwidth requirements are typically the parameters used to evaluate the QoS performance of a protocol. A real-time surveillance application, such as intruder detection/tracking requires very low latency; an environmental monitoring application, such as habitat monitoring using video, requires more bandwidth and is intolerant to jitter.

2.3.1.2 Energy efficiency

Energy efficiency is a key optimisation goal to extend the network lifetime. Energy efficiency entails the optimisation of protocols that directly affect the energy utilisation. The aim is to reduce the energy footprint of the sensor, processor, memory, transceiver, and other elements of a node. The radio transceiver has the largest energy footprint as the radio is used to perform the complex tasks of radio frequency (RF) communication. Various techniques must be used to minimise the energy usage of the transceiver. The focus on energy conservation and energy efficiency is extremely important to enhance the lifetime of the network.

2.3.1.3 Scalability

Since a WSN might include a large number of nodes, the architecture and protocols must be able to scale to these numbers. Some envisioned applications, such as Smart-Dust networks, have the potential to be of extremely large scale (tens of thousands of nodes). There are fundamental wireless limits on data throughput and channel capacity that impact the scalability and network performance.

2.3.1.4 Robustness

The vision of WSNs is to integrate many small inexpensive wireless nodes. However, inexpensive devices can often be unreliable and prone to failures. It is import to ensure that the global performance of the system is not sensitive to individual device failures. Protocol designs must have built-in mechanisms to provide robustness.

2.3.2 Key design factors

Even though there are numerous WSN application requirements, some design aspects influence all designs. A WSN design is influenced by many factors, which include: hardware constraints, fault tolerance, scalability, production costs, network topology, transmission media, and power consumption [1, 24].

2.3.2.1 Hardware constraints

The main concern for the operation of WSNs is energy consumption. Limited memory and processing power might be a determining factor for some designs, but the limited power source is the topic mostly covered when referring to hardware constraints and limited resources. The protocols designed for WSNs should be lightweight and the computational requirements of the algorithms low, to make up for the smaller processors and limited memory.

Most of the current WSN implementations use transceivers with low range, low bandwidth, and a low energy footprints; the complex tasks of RF communication is energy consuming and accounts for the largest part of energy expenditure in WSNs. These transceivers, which are more energy efficient than their high-speed IEEE 802.11 counterparts, still provide a serious limitation on network lifetime.

2.3.2.2 Fault tolerance

Since nodes are subject to hardware constraints and production costs limitations, the reliability of individual nodes are in question. Faults may occur because of a lack of power, physical damage, or the wireless link can be permanently interrupted; it is important that a WSN can tolerate such faults and not affect the overall performance of the network.

2.3.2.3 Scalability

The two main scalability factors are the large numbers of nodes and high-density deployments in WSNs. Since a WSN can consist of a large number of nodes the network architecture and protocols must be able to efficiently scale to these numbers. High-density deployments provide redundancy and improve fault tolerance but it adds additional strain on the limited bandwidth.

2.3.2.4 Production cost

The individual cost of sensor nodes must be very low to enable the envisioned large scale next-generation sensor networks. WSN devices are classified into two categories: data sinks or nodes. Nodes perform the sensing operation and are used for network construction which provide a transit path towards the sink. Data sinks act as gateways between a WSNs and traditional data networks. Additional features such as localisation systems, mobilisers, actuators, or energy harvesting systems are normally only found on data sinks. Protocols have been proposed to incorporate heterogeneous architectures to utilise the abilities of higher-end hardware.

2.3.2.5 Network topology

Network deployment is generally steered by coverage and connectivity objectives. Coverage pertains to the sensing quality of information required to perform the application-specific tasks. Connectivity pertains to the degree of connectedness and redundancy of wireless links. The network topology can be single-hop star topology, flat multi-hop mesh topology, structured placement topology or a multi-tier hierarchical cluster topology.

Multi-hop mesh and two-tier hierarchical cluster topologies have received the most interest from researchers. Flat multi-hop mesh topologies do not differentiate between the roles and capabilities of nodes. In hierarchical cluster schemes, the large network is divided into smaller zones which can be easier managed and allows for data aggregation. In random deployments, cluster-heads cannot be pre-determined and need to be determined by a selection process which adds additional complexity and resources requirements.

2.3.2.6 Transmission media

Wireless communication is the transmission media of choice but there are limitations between a sender and receiver pair based on distance; higher transmission power is required to communicate over longer distances. A multi-hop communication scheme is required which allows intermediate nodes to act as relays. Multiple short-range wireless communication links can be used together to reduce the total required power for transmissions. Popular options for radio links are the use of Industrial, Scientific and Medical (ISM) frequency bands, which offer licence free communication (such as the 430 MHz, 900 MHz and 2.4 GHz bands).

2.3.2.7 Power consumption

For flexibility in WSN deployments, devices are battery powered because of hardware constraints and production costs. The network lifetime is highly dependent on the battery lifetime. The basic requirements of a sensor node are sensing, processing, and data communication. The data communication accounts for the largest part of energy consumption in WSN devices.

2.4 WIRELESS NETWORK STANDARDS

The IEEE 802 Standards are used in order to provide a balance between the proliferation of a very large number of different and incompatible local and metropolitan networks, on the one hand, and the need to accommodate rapidly changing technology and to satisfy certain applications or cost goals on the other hand. The services and protocols specified in the IEEE 802 Standards map the lower two layers (physical and data link) of the seven-layer Open Systems Interconnection (OSI) communication system reference model.

Section 2.4.1 introduces the IEEE 802.15 Working Group (WG), Section 2.4.2 investigates the details of 802.15.4 and Section 2.4.3 describes the role of the ZigBee industry consortium.

2.4.1 IEEE 802.15 Working group

The 802.15 Wireless PAN effort focuses on the development of consensus standards for Personal Area Networks (PAN) or short distance wireless networks. These WPANs address wireless networking of portable and mobile computing devices such as PCs, Personal Digital Assistants (PDAs), peripherals, cell phones, pagers, and consumer electronics; allowing these devices to communicate and interoperate with one another.

There exists seven Task Groups (TGs) within the 802.15 Working Group (WG). Table 2.2 provides the details of these 802.15 TGs.

2.4.2 IEEE 802.15.4

The IEEE 802.15.4 standard has been widely accepted in the industry as the standard for enabling WSNs. It operates in the internationally unlicenced frequency bands. Potential applications include sensors, interactive toys, smart badges, remote controls, and home automation.

The 802.15.4 standard provides the following features: data rates of up to 250 Kbps; support for critical latency devices, such as joysticks; CSMA/CA channel access; automatic network establishment by the coordinator; fully handshaked protocol for transfer reliability; power management to ensure low power consumption; 16 channels in the 2.4 GHz ISM band, 10 channels in the 915 MHz and one channel in the 868 MHz band.

Even though the 802.15.4 provides a well-defined standard for WPANs it still lacks an important feature, time synchronisation, for superframes: ”In a peer-to-peer PAN,

TABLE 2.2: The IEEE 802.15 Task Group comparison.

Task Group	**Identification**	**Purpose**
802.15.1	WPAN/Bluetooth	Define a standard for WPANs based on the Bluetooth specifications.
802.15.2	Coexistence	Recommended Practices to facilitate coexistence of Wireless Personal Area Networks (802.15) and Wireless Local Area Networks (802.11).
802.15.3	High Rate WPAN	High data rate (20 Mbps or greater) WPANs for low power, low cost solutions addressing the needs of portable consumer digital imaging and multimedia applications.
802.15.4	Low Rate WPAN	Low data rate solution with multi-month to multi-year battery life and very low complexity.
802.15.5	Mesh networking	Determine the necessary mechanisms that must be present to enable mesh networking in WPANs.
802.15.6	Body Area Network (BAN)	Developing a communication standard optimised for low power devices and operation on, in or around the human body.
802.15.7	VLC	Creating a PHY and MAC standard for Visible Light Communications (VLC).

every device may communicate with every other device in its radio sphere of influence. In order to do this effectively, the devices wishing to communicate will need to either receive constantly or synchronise with each other. In the former case, the device can simply transmit its data using unslotted CSMA-CA. In the latter case, other measures need to be taken in order to achieve synchronisation. Such measures are beyond the scope of this standard" [26].

Table 2.3 shows the comparison between IEEE 802.11b, IEEE 802.15.1 and IEEE 802.15.4 based on different design criteria. The technical details for the 802.15.4 PHY can be found in Section 4.1.2, and the 802.15.4 MAC details can be found in Section 4.2.5.

2.4.3 ZigBee

The ZigBee wireless protocol standards are intended to provide monitoring, control, and sensory network services in a cost-effective manner that supports low data rates, low

TABLE 2.3: A comparison of wireless networking standards [27].

Market Name **Standard**	**Wi-Fi** IEEE 802.11b	**Bluetooth** IEEE 802.15.1	**ZigBee** IEEE 802.15.4
Type of Network	WLAN	WPAN	WPAN
Application Focus	Web, Email, Video	Cable Replacement	Monitoring and Control
System Resources	1MB+	250KB+	4KB - 32KB
Battery Life (days)	0.5 - 5	1 - 7	100 - 1000+
Network Size	32	7	255 / 65,000
Data rate (Kbps)	11,000+	720	20 - 250
Transmission Range (m)	1 - 100	1 - 10+	1 - 100+
Success Metrics	Speed, Flexibility	Cost, Convenience	Reliability, Power, Cost

power consumption, security, and reliable operation. The standards are communication stacks built on top of the IEEE 802.15.4 PHY/MAC Standard.

The ZigBee Alliance is a group of companies that maintain and publish the ZigBee standard. Over 300 leading semiconductor manufacturers, technology firms, Original Equipment Manufacturers (OEMs), and service companies comprise the ZigBee Alliance membership. The term ZigBee is a registered trademark of this group which is commonly mistaken as a single technical standard.

A key component of the ZigBee protocol is the ability to support mesh networking. In a mesh network, nodes are interconnected with other nodes so that multiple pathways connect each node. Connections between nodes are dynamically updated and optimised through sophisticated, built-in mesh routing protocols.

Mesh networks are decentralised in nature; each node is capable of self-discovery on the network. As nodes leave the network the mesh topology allows the nodes to reconfigure routing paths based on the new network structure. The characteristics of mesh topology and ad-hoc routing provide greater stability in changing conditions or failure at single nodes.

2.5 SIMULATORS

The emergence of wireless sensor networks created many open issues in network design. The three main traditional techniques for analysing the performance of wired and wireless networks were analytical methods, computer simulation, and physical measurement. However, many constraints imposed on sensor networks, such as power source limitation, decentralised collaboration, and fault tolerance necessitate the use of complex algorithms for sensor networks that usually defy analytical methods. It appears that simulation is currently the primary feasible approach to the quantitative analysis of sensor networks.

WSNs are rather complex, or even unfeasible, to model analytically and it usually leads to oversimplified analysis with limited confidence. Deploying test-beds involves a huge effort, adding financial project budget considerations and the incurring technical challenge to implement a test-bed. Therefore, simulation is essential to study WSNs. A suitable model based on solid assumptions and an appropriate framework to ease implementation is required. Simulation accuracy relies on the wireless channel and hardware modeling which are not usually accurate enough to capture the real world behavior of a WSN, thus, jeopardising the credibility of results. However, detailed models yield to scalability and performance issues due to the large number of nodes that have to be simulated. Therefore, the trade-off between scalability and accuracy becomes a major issue when simulating WSNs.

The selection criteria for viable WSN simulators under investigation are three-fold. The first is the availability of the simulator free of charge for academic use. The second is whether a vendor or community actively develops and supports the simulator or not. The third is the ability of the simulator to accommodate WSN technologies and topologies.

A survey on the most relevant network simulation environments used to study WSNs are provided in this section. The main features and implementation issues are described and discussed. The focus is on free, open-source network simulation tools with active development and support.

2.5.1 General purpose simulation packages

ns-2 [28] is a discrete event simulator targeted at networking research and is likely the most popular network simulator. The project started off as a general network simulator, and support for mobile ad-hoc wireless networks was added later. Drawbacks include not having a sensing model and a very simplistic energy model. ns-2 is not actively maintained at present due to the development of ns-3. The last release was in 2009.

ns-3 [29] is a new software development effort focused on improving upon the core architecture, software integration, models, and educational components of ns-2. ns-3 supports community involvement and ships a new stable version of ns-3 every three months, with new models developed, documented, validated, and maintained by enthusiastic researchers. Even though the project commenced in July 2006, WSN protocol implementations are still lacking. Development has been announced for IEEE 802.15.4 and 6LoWPAN by researchers [29].

OMNET++ [30] is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators. It is actively bringing out new releases (latest version released in May 2011) and multiple independent extension projects provide a wide variety of modeling frameworks. Popular extension projects for WSNs are Castalia and MiXiM which are described in Chapter 2.5.2.

J-Sim The fact that J-Sim is implemented in Java, along with its autonomous component architecture, makes J-Sim a truly platform-independent, extensible, and reusable environment [31]. The loosely-coupled component architecture of J-Sim enables the user to design, implement and test single components individually. New components can be easily added or exchanged for existing ones. However, choosing Java as the simulation language, inevitably sacrifices the efficiency of simulation for large-scale solutions. J-Sim is currently not actively developed as the last release and version 1.3 was released in 2004.

SSFNet [32] is a collection of components for the modeling and simulation of Internet protocols and networks at and above the IP packet level of detail. It is based on the Scalable Simulation Framework (SSF) discrete-event simulator. SSFNet does not include physical or link layer components, and the last release was in 2004.

GloMoSim [33] is a network protocol simulation software that simulates wireless and wired network systems. It is built upon a parallel discrete-event simulator framework. GloMoSim follows the idea of the OSI reference model by using a layered approach. The communication between the different simulation layers uses a standard Application Programming Interface (API) so that new models and layers can be rapidly exchanged and integrated. Development stopped in 2000 and work continued on its commercial derivative QualNet (which is not open-source or for academic use).

2.5.2 WSN Specific Simulation Tools

Castalia [34] is a WSN simulator for networks of low-power embedded devices. It is an OMNeT++ platform extension and can be used to test distributed algorithms and protocols using realistic wireless channel and radio models. The models were created using empirical measurement data and simulates realistic radio access, processing, sensing, and energy usage. Castalia has an active online community and released a new version in March 2011.

SensorSim [35] inherits the core features of traditional event driven network simulators. It uses new features that include the ability to model power usage in sensor nodes, hybrid simulations that allows the interaction of real and simulated nodes, and new communication protocols and real time user interaction with graphical display. The simulator was never released for public use and development has stopped.

TOSSIM captures the behavior and interactions of networks of thousands of TinyOS motes at network bit granularity [36, 37]. TOSSIM is a very powerful emulator which allows users to debug, test, and analyse algorithms in a controlled and repeatable environment. The primary goal of TOSSIM is to simulate TinyOS and its execution, rather than specialising in simulating the real world. It requires knowledge of the component based event-driven programming language, NesC.

MiXiM [38] is an OMNeT++ extension modeling framework created for mobile and fixed wireless networks. MiXiM concentrates on the lower layers of the protocol stack and offers detailed models of RF propagation, interference estimation, radio transceiver power consumption and wireless MAC protocols. The lack of proper documentation and zero references in highly-cited WSN research articles is a serious drawback. A new version of MiXiM was released in June 2011.

ATEMU is one of the first instruction-level software emulators for AVR (8-bit microcontroller by Atmel) based systems. Although ATEMU is the most accurate instruction-level emulator for wireless sensor network research, it lacks simulation speed, being 30 times slower than TOSSIM [39]. The last revision change was introduced in 2006.

Avrora is a set of simulation and analysis tools for programmes written for AVR micro-controllers [40]. The special characteristic of Avrora is that it operates on the instruction-level which provides an accurate simulation of devices and radio communication. Avrora uses event-queues to increase scalability performance (over emulators such as TOSSIM and ATEMU). Avrora lacks an integrated Graphical User Interface (GUI) and last development activity was in 2007.

EmStar is a software environment for developing and deploying complex applications on heterogeneous networks. EmStar is designed to leverage microservers (nodes which are less constrained in computational power and data storage size) by trading off some performance for system robustness in sensor network applications [41]. It enables fault isolation, fault tolerance, system visibility, in-field debugging, and resource sharing across multiple applications. EmStar services include support for networking, sensing, and time synchronisation. EmStar is currently not actively developed; the last release was in 2005.

2.6 SUMMARY

This chapter has introduced the basic concept and high-level overview of wireless sensor network devices. The basic sensor node hardware building blocks were identified as the embedded processor, memory, radio transceiver, sensor, and a power source. There are many proposed and implemented applications for WSNs which include environmental monitoring, building automation, health applications, logistics, disaster relief applications, facility management, machine surveillance and preventive maintenance, precision agriculture, and smart transportation.

The type of application determines the optimisation goals and design factors. The design optimisation goals were classified into quality of service, energy efficiency, scalability, and robustness. A WSN design is influenced by many factors, which include: hardware constraints, fault tolerance, scalability, production costs, network topology, transmission media and power consumption.

The IEEE 802.15 Working Group focuses on consensus standards for Wireless Personal Area Networks (WPAN) and consists of seven task groups. The Low Rate WPAN (IEEE 802.15.4) standard has been widely accepted in the industry as the standard for enabling WSNs. The ZigBee Alliance is a group of companies that maintain and publish the ZigBee wireless protocol standards. The ZigBee standards are built on the IEEE 802.15.4 PHY/MAC and provide the network, transport and application layer functions.

The survey on free, open-source, simulation tools with active development and support yielded Castalia as the WSN simulator of choice for the proposed research. Castalia is used to test distributed algorithms and protocols using realistic wireless channel and radio models. The models were created using empirical measurement data and simulates realistic radio access, processing, sensing, and energy usage.

CHAPTER 3

SLEEP SCHEDULING

Objectives of this Chapter

The radio transceiver is the hardware component that uses the most power in a WSN node and the optimisation of this element can reduce the total energy consumption. The purpose of sleep scheduling techniques is to save energy and prolong network lifetime. In this chapter sleep scheduling and surrounding technologies are reviewed. After this chapter the reader should have an understanding of the fundamentals of sleep scheduling and the effect it has on the network lifetime. Time synchronisation and clock drift concepts are introduced.

3.1 RADIO OPTIMISATION EFFECT

It is essential to model the power demands of a WSN node to efficiently optimise the energy consumption. This section models the main functional components that contribute to the energy consumption of a TelosB WSN node. The effectiveness of duty cycled sleep scheduling schemes on the radio is investigated.

In the work of [42], a model for the TelosB WSN node was generated based on empirical data. The application-specific sensing unit contribution was excluded from their model. The energy model defined a duty cycled (DC) model for the CPU, radio, and Light Emitting Diodes (LEDs) using measured current data:

$$\begin{aligned} E[mA] &= (2.33 \cdot DC_{CPU\_ACTIVE} + 2.25 \cdot DC_{CPU\_IDLE} + \\ &\quad 22 \cdot DC_{RADIO\_ON} + 2.4 \cdot DC_{RADIO\_OFF} + \\ &\quad 3.8 \cdot DC_{RED} + 3.3 \cdot DC_{BLUE} + 5.3 \cdot DC_{GREEN} + \\ &\quad E_{FLASH} \cdot FREQ_{FLASH}) \cdot T_{ALL} \end{aligned}$$

From their measurements, the current drawn for a 4 MHz CPU was 2.33 mA in active mode and 2.25 mA in idle mode, while operating at 1.80 μA in disabled mode. The internal flash memory access draws between 1.9 mA and 2.3 mA, but the final effect depends on the reprogramming frequency ($FREQ_{FLASH}$). The LEDs' power usage is hardly ever mentioned in energy consumption models, but they draw 3.9 mA, 3.3 mA and 5.3 mA respectively for the red, blue and green LEDs. In WSN deployments without human intervention the use of the LEDs are void and can be completely ignored.

The radio transceiver related power consumption is the most critical component of the system-wide power consumption. In the receive/listening state the current drawn is 22.8 mA, 21.7 mA (at 0 dBm) in the transmitting state and only 2.4 mA when the radio is off (microcontroller (MCU) still on). In this model the effect of the sensing unit is not included as it is highly dependent on the application (which type of sensor, sampling period, etc.).

In Figure 3.1 it is clear that sleep scheduling can have a major effect on the overall energy consumption. The effect of the DC_{RADIO_OFF} (2.4 mA) was removed from the presented model because the transceiver uses 1 μA in the sleep state [43] and the MCU effect is already included by the DC_{CPU_ACTIVE} variable. Using the parameters set in Figure 3.1, the current drawn when $DC_{RADIO\_ON} = 100$ % is 24.935 mA and only 3.155 mA when $DC_{RADIO\_ON} = 1$ %. This shows the major improvement that can be achieved by only implementing ultra-low duty cycle sleep scheduling.

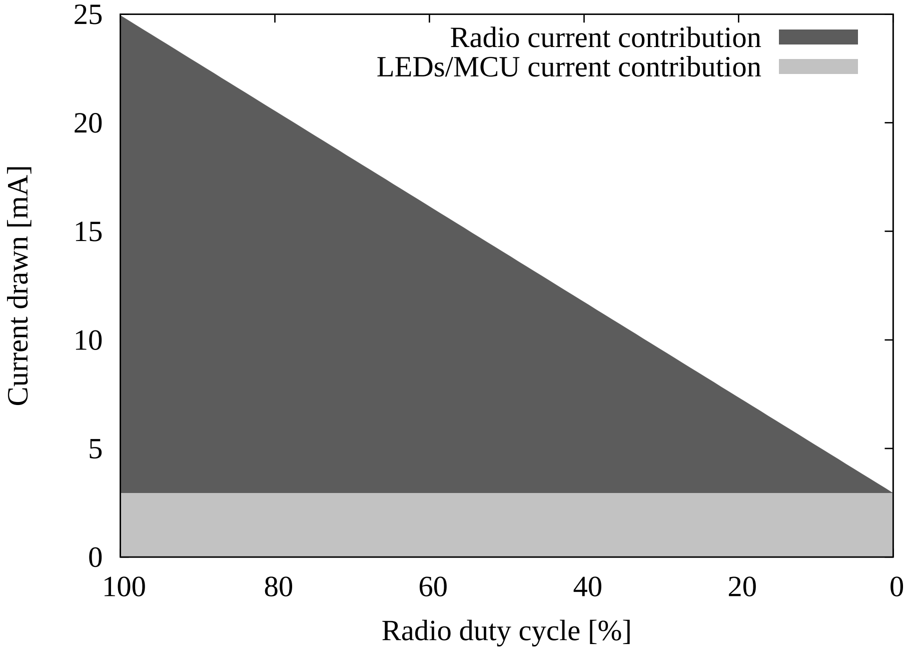


FIGURE 3.1: Radio sleep scheduling effect on TelosB WSN node drawn current. $DC_{RED} = DC_{BLUE} = DC_{GREEN} = 5\%$, $DC_{CPU\_ACTIVE} = 100\%$, $DC_{CPU\_IDLE} = 0\%$ and $FREQ_{FLASH} = 0$

3.2 SLEEP SCHEDULING TECHNIQUES

The purpose of sleep scheduling techniques is to save energy and prolong network lifetime. Several protocols have been proposed to reduce the energy consumption using sleep scheduling methods. This section describes the fundamental asynchronous and scheduled sleep techniques.

3.2.1 Asynchronous sleep techniques

Asynchronous sleep techniques aim to keep the radio in default sleep mode and wake-up briefly to check for traffic or send/receive messages. The following asynchronous sleeping techniques are discussed: secondary wake-up radios, low-power listening/preamble sampling, WiseMAC and TICER/RICER.

The *secondary wake-up radio* concept uses a hardware solution to enable nodes to sleep by default and only awake when needed [4]. In the design, each sensor node is equipped with two radio transceivers. The primary (data) radio remains asleep by default. The secondary radio is a low-power wake-up radio that remains on at all times. When the secondary radio receives a wake-up signal from another node, it instructs the primary radio to wake up for data transmission. This method assumes that an extremely low-power radio can be used as a secondary radio. This method does not scale well for densely deployed WSN topologies; all nodes within the broadcasting range will suffer from overhearing, waking up for unicast packets which might not be destined to them.

In *preamble sampling* [5] a node will transmit a preamble to indicate that it has data to send, as shown in Figure 3.2. All nodes will periodically wake up and sense whether a preamble is present. If no activity is found the node can go back to sleep. Otherwise, a node will stay in active mode and wait for the sender node to start transmission of the data. This scheme also suffers from overhearing problems.

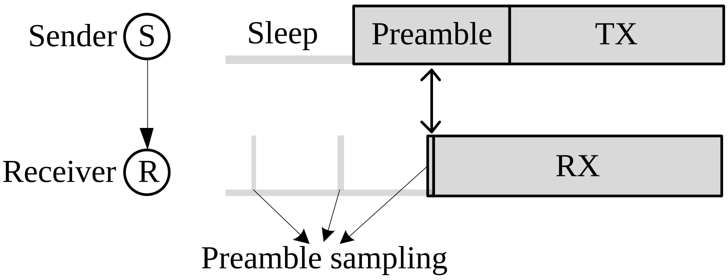


FIGURE 3.2: The low-power listening technique of preamble sampling.

The TICER/RICER [7] techniques are similar to the low-power listening/preamble sampling but include additional features. In the *transmitter-initiated cycle receiver (TICER)* technique (see Figure 3.3), the sender periodically sends a request to send (RTS) signal to indicate a pending transfer. The receiver wakes up periodically to monitor the channel and will respond with a clear to send (CTS) signal; data transmission will begin immediately after the receiver responded to the RTS.

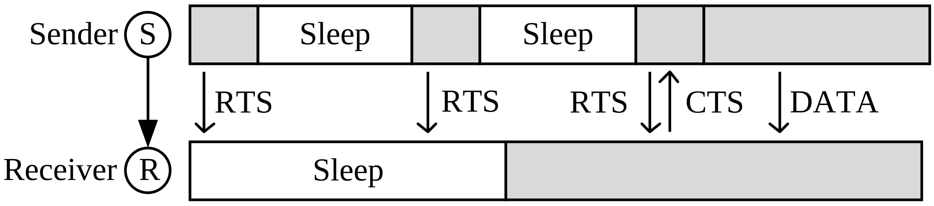


FIGURE 3.3: Transmitter-initiated cycle receiver (TICER).

The *receiver-initiated cycle receiver (RICER)* (see Figure 3.4) differs slightly from TICER, by having the receiver periodically transmit beacons. A source that wishes to send data, wakes up and listens until it receives a beacon message; the sender begins transmissions after it received the beacon message. The drawback of the TICER/RICER methods is that they do not actively learn about the periodic schedules of other nodes.

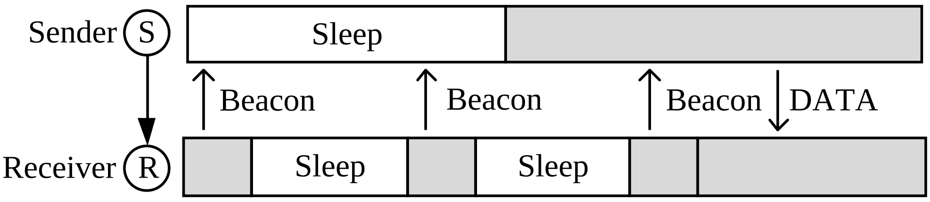


FIGURE 3.4: Receiver-initiated cycle receiver (RICER).

The *WiseMAC* [6] protocol builds on the shortcomings of the preamble sampling technique: reducing the long preamble which can cause throughput reduction and energy wastage for both sender and receiver. Nodes learn about the periodic sampling times of its neighbouring nodes and use this information to reduce the wake-up preamble length, which saves energy. The protocol also includes the necessary adjustments to cater for clock drift.

Asynchronous sleep techniques provide low-complexity solutions; the trade-off for low-complexity solutions is their efficiency. These solutions do not scale well to large nodal populations and densely deployed WSNs and more effective energy management can be implemented by scheduled sleep patterns.

3.2.2 Sleep-scheduled techniques

Sleep scheduling techniques aim to reduce energy consumption by synchronising sleep schedules and enable lower duty cycles. S-MAC, T-MAC, and D-MAC are the most well-known WSN sleep scheduling techniques.

Sensor MAC (S-MAC) [44] provides a tuneable, non-adaptive, periodic active/sleep cycle for sensor nodes (see Figure 3.5). During sleep periods nodes turn off their radios to conserve energy. During active periods nodes turn on their radios to exchange data.

During the initialisation phase nodes remain awake and listen for sleep schedules from neighbours. If they do not receive a sleep schedule, they create their own schedule and start broadcasting it. A problem with S-MAC is that a single node can follow multiple schedules, which can severely impact the average duty cycle. S-MAC includes RTS/CTS handshaking for collision avoidance and virtual carrier sensing using a network allocation vector (NAV) for overhearing avoidance.

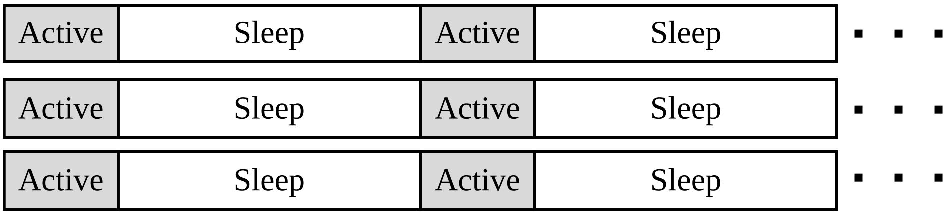


FIGURE 3.5: Sleep scheduling in S-MAC.

Timeout MAC (T-MAC) [9] includes all the features of S-MAC, but additionally includes an adaptive listening scheme which adapts the active portion of each cycle dynamically, as shown in Figure 3.6. T-MAC uses a timeout mechanism to adapt the length of the active period. If a node does not receive any messages during the timeout interval, it goes to sleep; if a node receives a message, it resets the timeout interval. The active period ends when no activation event has occurred for a certain time. This method reduces idle listening compared to S-MAC, while allowing an adaptive active period needed during higher data throughput bursts.

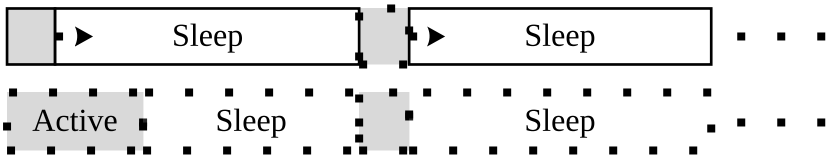


FIGURE 3.6: Sleep scheduling in T-MAC.

Figure 3.7 shows the different timeout methods [10] that can be used in T-MAC. A new active period starts at t_0. At t_1 a transmission is detected which continues until t_2. The ideal scenario is depicted in scenario A: the activation timeout would reset at the end of transmission. Scenario B indicates the least-effort approach, the timeout is reset at the start of transmission. In Scenario B, knowledge of the transmission duration is not necessary. Scenario C is a conservative approach, no communication should have taken place for at least one timeout duration. The research showed that scenario C has the best packet reception rate, but consumed more energy because of the longer active time. A problem in T-MAC occurs when a node cannot transmit an activation event to its intended receiver because of medium contention and the receiver goes to sleep. This early sleep problem reduces throughput and increases latency.

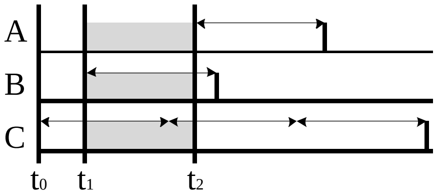


FIGURE 3.7: T-MAC activation timeout methods.

The data forwarding interruption (DFI) problem exists in duty-cycle techniques where the radio overhearing range is limited. Nodes that are outside of the communication range of both the sender and receiver are unaware of ongoing data transmission, and therefore sleep until the next cycle. *Data-gathering MAC (D-MAC)* [11] was proposed to combat the DFI problem. D-MAC delivers data along a data gathering tree, aiming at both energy efficiency and low latency. Figure 3.8 shows a data gathering tree that forms part of the tree-style source-to-sink data delivery path. It shows the staggered schedule of the nodes where the transmission aligns with the reception portion of the parent node. Nodes are divided based on the level in the data gathering tree and the structure allows for continuous packet forwarding.

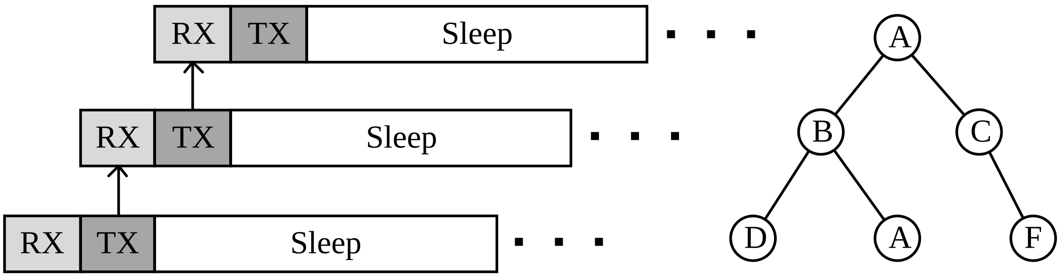


FIGURE 3.8: Sleep scheduling in D-MAC.

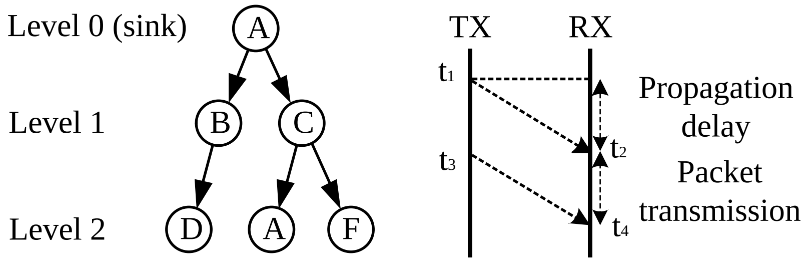
3.3 TIME SYNCHRONISATION

There are several reasons for addressing the synchronisation problem in WSNs. Firstly, synchronisation can be used by sleep scheduling algorithms to increase the network lifetime. Nodes need to coordinate their sleep schedules, awake at precise timings, and sleep at appropriate times to conserve energy.

Secondly, for sensors to generate valuable information it needs to provide timestamps. For example, a fleet monitoring application can record the movement of vehicles. Using this information, vehicles can be actively tracked and the collected data can be used to provide optimised traveling routes for the entire fleet.

The allocation of time slots in a sleep scheduling protocol is a complex problem that requires coordination and synchronisation. Sensor nodes need to agree on set boundaries for the time slots, otherwise transmissions would overlap and result in collisions. The need for low-cost and energy efficient time synchronisation solutions are clearly required.

In the *network time protocol (NTP)* [45], a stratum 0 device indicates an atomic clock. A stratum level 1 indicates that it synchronises with an atomic clock. An hierarchical structure is followed where each time server of each level synchronise the clocks of their sub-level peers. Similar to this the hierarchical synchronisation architecture scheme (see Figure 3.9(a)) for WSNs is presented. The sink node acts as the reference node for the whole network and would synchronise itself with an external source.



(a) Hierarchical synchronisation architecture. (b) Single message time synchronisation.

FIGURE 3.9: Time synchronisation using a tree-structure and sender-only method.

The single message approach is a very lightweight synchronisation algorithm with minimal complexity but very low precision. When using a single message approach, two measurements need to be taken into account, as shown in Figure 3.9(b): the propagation delay and the transmission time. The propagation delay is the time it takes for the message to travel from the sender to the receiver. The propagation delay is almost negligible, as light travels the distance of 30 metres in 0.1 μs. The packet transmission time is a function of the size of the synchronisation packet and the transceiver data rate. This method requires periodic synchronisation, in contrast to client initiated methods, and only the data sink needs an external clock reference. The re-synchronisation frequency is determined by the clock drift and data gathering tree depth.

The two-way message handshake technique presented in NTP provides higher precision. This method is the basic building block of many synchronisation protocols for sensor networks, for example, the *timing-sync protocol for sensor networks (TPSN)* [46]. The handshaking method takes the following components into account: sending delay, access delay, propagation delay and receiving delay. The client initialises the procedure by sending a synchronisation request to its server and the server replies with an acknowledgement. The client node calculates the clock offset and propagation delay and then synchronises itself to the server.

The *flood time synchronisation protocol (FTSP)* [47] performs network-wide synchronisation using a single message per synchronisation. FTSP implements MAC layer timestamping and use controlled flooding to synchronise the network. A node collects multiple timestamps and performs synchronisation by estimating its clock drift and offset using linear regression.

3.4 CLOCK DRIFT

Clock drift refers to a phenomena where a clock has a small random deviation from its nominal frequency. An analogy can be made by comparing the clock drift effect to the situation where your wristwatch can be off a few minutes each year, and you have to compensate for this by setting it back to the reference or correct time. Time synchronisation is complicated even further by clock drift and more frequent synchronisation is required to compensate for the clock drift effect.

Figure 3.10 shows the clock drift effect for a fast and a slow clock in relation to a reference clock. At time t_0 all three clocks are synchronised. All clocks should stop after $T = (t_1 - t_0)$. The fast clock stops at t_2, which is off by $\Delta t_{FAST} = t_1 - t_2$. The slow clock stops at t_3, which is off by $\Delta t_{SLOW} = t_3 - t_1$. When you compare the slow clock in relation to the fast clock, the difference is $\Delta t_{MAX} = \Delta t_{SLOW} + \Delta t_{FAST} = t_3 - t_2$. The worst-case scenario is when the time difference is maximised: one clock deviates at the lower error limit and one clock deviates at the upper error limit.

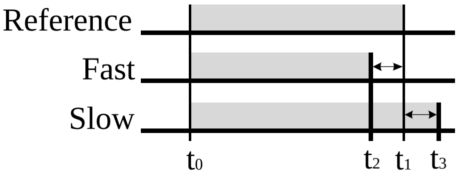


FIGURE 3.10: The timing effect of clock drift.

The S-MAC protocol requires the synchronisation of sleep schedules between nodes; the S-MAC protocol scenario illustrates the negative impact of clock drift, see Figure 3.11. The assumption is made that both the sender and receiver nodes are synchronised at the start. The sender node has a fast clock and the receiver node a slow clock. After two sleep cycles the nodes have lost sleep synchronisation and will not be able to communicate anymore. Lost synchronisation has devastating effects in sleep scheduling protocols which increase packet loss, latency, and jitter.

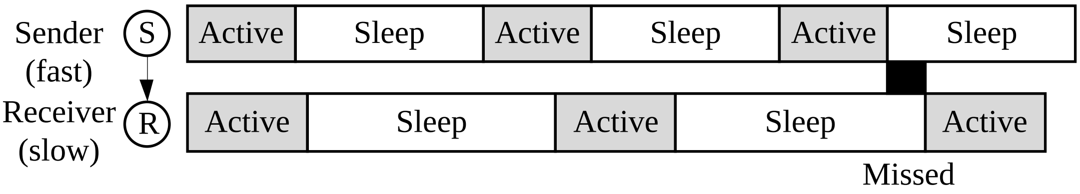


FIGURE 3.11: Synchronisation errors caused by clock drift during sleep scheduling in S-MAC.

The TelosB WSN node uses two different oscillator crystals: a 32 kHz crystal and a 16 MHz crystal. The clock module in the MSP430F1611 (the MCU used in the

TelosB node) provides three clock signals: auxiliary clock sourced from the 32 kHz crystal, main clock (system clock) used by the CPU, and the sub-main clock used by peripheral modules. The system clock is in control of the MCU timing. The second oscillator provides the 16 MHz reference frequency for the CC2420 transceiver module.

In [48] a 32 kHz crystal oscillator selection criteria guide for the MSP430 microcontroller is presented. The application report specifies the crystal tolerance, which should be between 5 parts per million (ppm) and 30 ppm. If considered a clock drift of at most $\pm\delta$ per second, the actual wake-up time of a node may drift by $T_{MAX} = \pm 2\delta T$ (where T represents the time until the next event). A deviation of 30 ppm amounts to 80 μs maximum error every 1 s, 0.8 ms every 10 s and 4.8 ms every 60 s.

3.5 SUMMARY

This chapter reviewed the effect of duty cycling the radio can have on energy consumption and the design considerations introduced by clock drift. Using an energy model for the TelosB node, a 87 % saving in the energy consumption was achieved by introducing a 1 % duty cycle on the radio. The improvement utilised the low-power state of the radio and translated into an increase in nodal lifetime of 790 % (7.9 times improvement).

The literature review revealed two types of sleep techniques: asynchronous sleep techniques and scheduled sleep techniques. The scheduled techniques outperformed the asynchronous techniques; the sleep scheduled techniques provided better performance by synchronising the schedules of nodes. Sleep scheduling techniques improved bandwidth, latency, and reliability but added additional protocol overhead, and required time synchronisation for scheduling.

Clock drift is an important real-world effect and it is important to compensate for this error. A deviation of 30 ppm amounts to 0.8 ms maximum error every 10 s, 2.4 ms every 30 s, and 4.8 ms every 60 s. It is important to compensate for clock drift to ensure desynchronisation does not occur.

CHAPTER 4

COMMUNICATION PROTOCOLS

Objectives of this Chapter

This chapter is devoted to communication protocols used in WSNs. Each layer of the WSN protocol stack is presented and described. The focus of this chapter is on the data link and network layer. A brief overview of the physical and application layer is given. At the end of this chapter the reader should have an understanding of the WSN protocol stack, the different functions of each layer, and different protocols, algorithms, and techniques used.

Unlike other types of communication networks, the traffic patterns of WSNs are simple and predictable. Most data-centric WSNs are described by traffic patterns with unidirectional flow from sensor nodes to the data sink, with occasional network control packets flowing downstream.

The protocol stack used by all nodes in the network is given in Figure 4.1. It is a very lightweight stack and does not go into as much detail as the stack by Akyldiz et al. [1]. The most prominent difference is the lack of the transport layer presented in the proposed model. The transport layer is responsible for providing reliable data transfer services and congestion control. However, for periodic sampling applications the bandwidth utilisation is not expected to be high and congestion control is not necessary. Also, reliability can be provided in the data link layer using acknowledgements (ACK) and automatic repeat request (ARQ). This allows for the removal of the transport layer in the presented protocol stack which reduces the per-packet overhead.

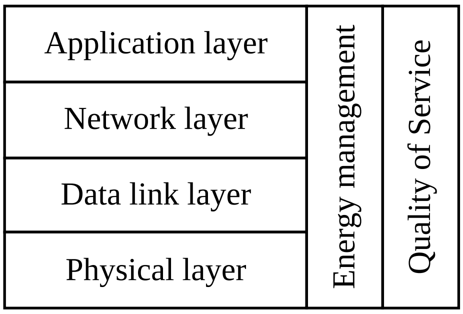


FIGURE 4.1: The sensor network protocol stack.

The physical layer is responsible for the conversion of the bitstream into signals that are best suited for communications across the wireless channel. The data link layer ensures communication in the wireless medium so that the communication links between nodes are established and connectivity is provided throughout the network. The network layer performs the important task of establishing and maintaining routing tables, and route data across the network to establish multi-hop communication routing. The application layer is the heart of the application, as it manages time synchronisation, localisation and sensing tasks. Energy management and quality of service are cross-layer optimisation goals which should be taken into consideration when designing protocols across all layers.

4.1 PHYSICAL LAYER

The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, modulation, and data encryption. Even though there is a large body of literature that can be included in this section, this is beyond the scope of the dissertation and a short review of physical layer technologies are given.

4.1.1 Classification of PHY methods

Figure 4.2 shows the physical layer technology breakdown. The main types of RF access technologies used in WSNs can be classified into: narrowband, spread spectrum, and ultra wideband (UWB) techniques. The RF data techniques include channel coding and modulation/interleaving.

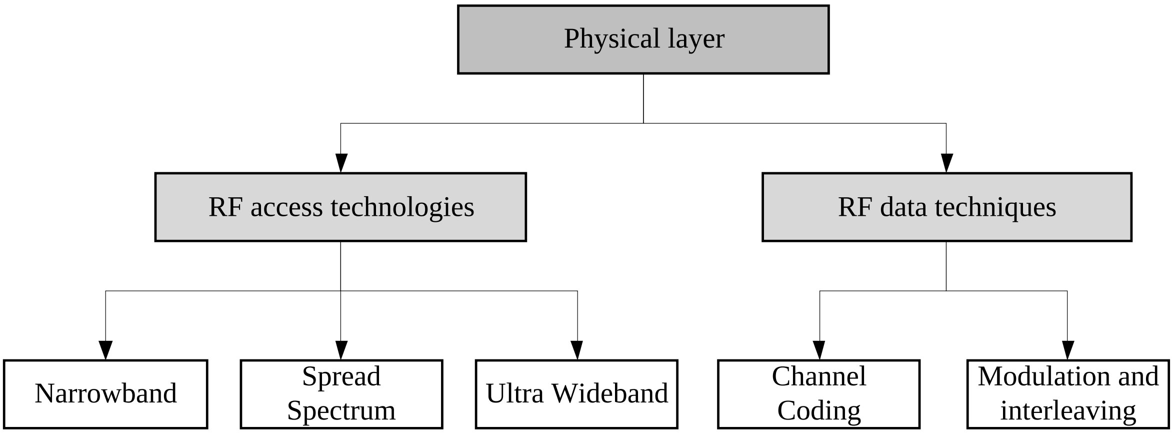


FIGURE 4.2: Physical layer technology breakdown.

4.1.1.1 RF communications technologies

Narrowband systems support low bit-rate transmission, whereas wideband systems support high bit-rate transmission. A system is defined as narrowband or wideband depending on the bandwidth of the transmission physical channel with which it operates. The system channel bandwidth is assessed with respect to the coherence bandwidth. The coherence bandwidth is defined as the frequency band within which all frequency components are equally affected by fading due to multipath propagation. Systems operating with channels narrower than the coherence bandwidth are known as narrowband systems. Wideband systems operate with channels wider than the coherence bandwidth.

In *spread spectrum* systems a narrowband signal is transmitted using a spectrum that is much larger than the frequency content of the signal. This technique decreases the potential interference to other receivers while achieving privacy. Frequency-hopping spread spectrum (FHSS), direct-sequence spread spectrum (DSSS), time-hopping spread spectrum (THSS), chirp spread spectrum (CSS), and combinations of these techniques are different forms of spread spectrum technologies. The CC2420 transceiver used by many WSN platforms uses DSSS.

Ultra wideband (UWB) communication is fundamentally different from all other communication techniques because it employs extremely narrow RF pulses to communicate between transmitters and receivers. Utilising short-duration pulses as the building blocks for communications directly generates a very wide bandwidth and offers several advantages, such as large throughput, robustness to jamming, coexistence with current radio services, a simple transceiver architecture, low transmit power, and high performance in multipath channels.

4.1.1.2 RF communication techniques

Channel coding deals with error control. If data at the output of a communication system has errors that are too frequent for the desired use, the errors can often be detected, reduced or corrected. The two main methods of error control are automatic repeat request (ARQ) and forward error correction (FEC). ARQ requests a packet retransmission when errors are detected. FEC adds redundant data to its messages, which allows for error detection/correction.

Modulation is the process of converting a digital bitstream into an analog signal which contains the information. The speed of a transmission can be correlated to the symbol rate and the number of bits per symbol. There are three fundamental modulation types: Amplitude Shift Keying (ASK), Phase Shift Keying (PSK), Frequency Shift Keying (FSK), or a combination of them. Any digital modulation scheme uses a finite number of distinct signals to represent digital data. Binary digits are then assigned to unique symbols which can be used for modulation.

Interleaving is used in digital communication to improve the performance of forward error correcting codes. In communication systems, errors typically occur in bursts.

Error-correcting codes can only recover the original code word if the number of errors do not exceed the code's capability. By shuffling the bitstream, errors are more uniformly spread and increases the ability to recover from errors caused by bursts.

4.1.2 IEEE 802.15.4 PHY

The IEEE 802.15.4 PHY is responsible for the following tasks: activation and deactivation of the radio transceiver; energy detection (ED) within the current channel; link quality indication (LQI) for received packets; clear channel assessment (CCA) for CSMA/CA; channel frequency selection; and data transmission and reception.

An IEEE 802.15.4 compliant device shall operate in one or several frequency bands using the modulation and data parameters given in Table 4.1.

TABLE 4.1: IEEE 802.15.4 PHY Frequency bands and data rates [26].

PHY (MHz)	**Frequency band (MHz)**	**Modulation**	**Bit rate (Kbps)**	**Symbol rate (Ksymbol/s)**	**Symbols**
868/915	868-868.6	BPSK	20	20	Binary
	902-928	BPSK	40	40	Binary
2450	2400-2484.5	O-QPSK	250	62.5	16-ary Orthogonal

4.1.3 Existing transceivers

The CC2420 is a true single-chip 2.4 GHz IEEE 802.15.4 compliant RF transceiver designed for low-power and low-voltage wireless applications. The low current consumption (RX: 19.7 mA, TX: 17.4 mA) makes it an ideal transceiver for WSN applications which require extremely low power solutions. The Texas Instruments CC2420 chip is used by such platforms as TelosB, MicaZ, SunSPOT and Imote2. A list of existing transceivers is given in Table 4.2.

TABLE 4.2: Existing transceivers used in WSN [49].

	Radio				
	RFM TR1000	**Infineon TDA5250**	**TI CC1000**	**TI CC2420**	**Zeevo ZV4002**
Platforms	WeC, Rene, Dot, Mica	eyesIFX	Mica2Dot, Mica2, BTnode	MicaZ, TelosB, SunSPOT, Imote2	Imote, BTnode
Standard	N/A	N/A	N/A	IEEE 802.15.4	Bluetooth
Data rate (Kbps)	2.4-115.2	19.2	38.5	250	723.2
Modulation	OOK/ASK	ASK/FSK	FSK	O–QPSK	FHSS-GFSK
Radio frequency (MHz)	916	868	315/433/868/915	2.4 GHz	2.4 GHz
Supply voltage (V)	2.7-3.5	2.1-5.5	2.1-3.6	2.1-3.6	0.85-3.3
TX max (mA/dBm)	12/-1	11.9/9	26.7/10	17.4/0	32/4
TX min (mA/dBm)	N/A	4.9/-22	5.3/-20	8.5/-25	N/A
RX (mA)	1.8-4.5	8.6-9.5	7.4-9.6	18.8	32
Sleep (μA)	5	9	0.2-1	0.02	3.3mA
Startup time (ms)	12	0.77-1.43	1.5-5	0.3-0.6	N/A

4.2 DATA LINK LAYER

The data link layer provides addressing and channel access control mechanisms that make it possible for several network nodes to communicate reliably within a point-to-point or point-to-multipoint network. The medium access control (MAC) data communication protocol sub-layer, also known as the media access control, is a sublayer of the data link layer. A fundamental task of the wireless MAC protocol is to avoid collisions so that two interfering nodes do not transmit at the same time. The design objectives for low-energy WSN MAC protocols differ completely from MAC protocols for traditional wireless computer networks, such as WiFi (IEEE 802.11). While the latter pursue to maximise the achieved throughput, low-energy WSN MAC protocols aim to maximise energy-efficiency and network lifetime.

The MAC protocol in a wireless multi-hop self-organising sensor network must achieve two goals. Firstly, the creation of the network infrastructure. Since thousands of sensor nodes are densely scattered in a sensor field the MAC layer must establish communication links for data transfer. This forms the basic single-hop infrastructure needed for wireless communication and creates a self-organising network. The second objective is to fairly and efficiently share communication resources between sensor nodes. WSNs have unique resource constraints which emphasise the fact that novel protocols and algorithms are needed to address these issues. The broadcast nature of the wireless channel requires the MAC to coordinate channel access and limit energy wastage caused by collisions, idle listening, network signalling overhead, overhearing and over-emitting.

This section provides the classification of MAC methods, a review on the fundamentals of MAC protocols for WSNs, and a study on the different MAC techniques and protocols currently proposed for WSNs.

4.2.1 Classification of MAC methods

Figure 4.3 shows a brief overview of the most common medium-access control methods. The classification divides MAC methods into fixed-assignment, demand-assignment, and contention-access protocol categories. The list is not exhaustive and there are many other medium access control methods not included [50]. Consideration should be taken, a MAC protocol might not simply fit into a single category: a system can use reservation methods for signalling on a given channel and CSMA for data transfer on another channel.

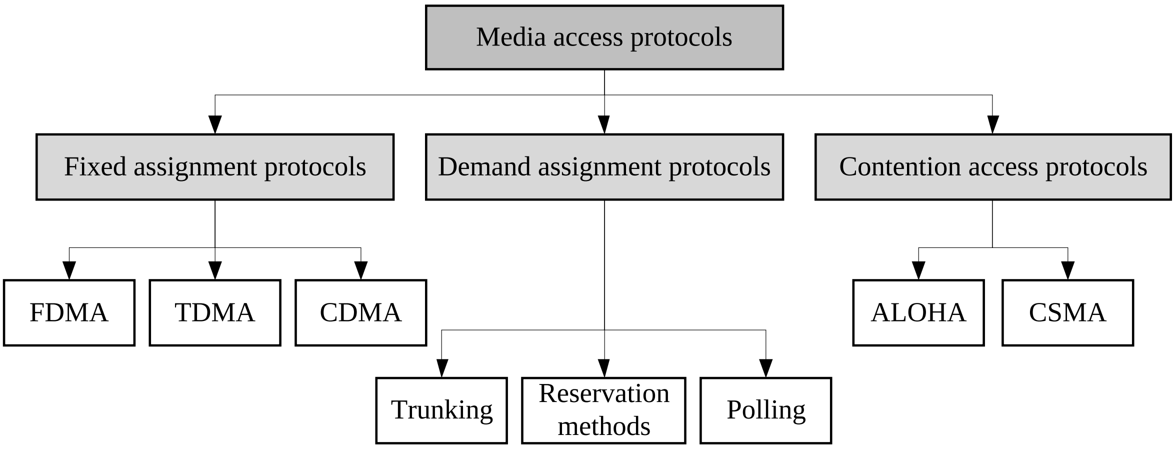


FIGURE 4.3: Medium-access protocol taxonomy.

4.2.1.1 Fixed-assignment

In fixed-assignment protocols nodes are assigned unique time slots, frequency channels, or spreading codes for transmissions, eliminating collisions. This simplifies individual transmissions but the bandwidth must be reserved prior to data transmissions, increasing signalling overhead while trying to avoid frame collisions.

Frequency Division Multiple Access (FDMA) allocates one or several frequency bands (channels) to clients. The coordination tasks play an important role in FDMA systems promoting better spectrum utilisation. By assigning fixed frequencies for all potential users, poor spectrum utilisation could result if users do not intend to use the allocated bandwidth (unoccupied channels). Guard bands are introduced to provide spacing between user channels; guard bands are used to reduce crosstalk which causes interference. A trade-off must be made between spectrum efficiency and the cost of filtering: smaller guard bands increase spectrum efficiency but also increases the associated cost.

Time Division Multiple Access (TDMA) based protocols are naturally energy preserving, because they have a built-in duty cycle and do not suffer from collisions. However, maintaining a TDMA schedule in an ad-hoc network is not an easy task and requires complex coordination between nodes. Keeping a list of neighbour schedules uses valuable memory. The allocation of TDMA slots is a complex problem that requires coordination. Furthermore, TDMA divides time frames into small slots and the effect of clock drift can be disastrous; exact timing is critical.

Code Division Multiple Access (CDMA) employs spread-spectrum PHY layer technology and a special coding scheme (each transmitter is assigned a unique code) to allow multiple users to be multiplexed over the same physical channel. There are two types of spread spectrum used by CDMA: frequency hopping and direct sequence. Frequency hopping switches the carrier frequency of the modulated signal from channel to channel in a pseudorandom pattern (code). Direct sequence keeps the carrier frequency constant and multiplies the transmitted binary data by a predetermined, high-frequency, pseudorandom spreading code prior to modulation of the carrier.

4.2.1.2 Demand-assignment

Demand-assignment protocols attempt to improve on the channel inefficiencies of fixed-assignment protocols, by reassigning unused channel assets to users that can use them. In general, this requires the use of a controller to arbitrate between users, and makes demand assignment protocols more complex than their fixed counterparts, since needy users and available channel assets must be matched. It also generates the need for a logical control channel, separate from the logical data channel over which messages are passed.

Trunking is a multiple-access scheme that dynamically assigns communication requests to available logical channels. Any fixed-assignment MAC protocol can be employed, with the goal to substantially improve the channel efficiency without causing degradation of QoS to any user. When devices are not sending traffic, it monitors the outbound control channel. When a device has a message for a specific group, it sends a request on the inbound control channel. The trunking system identifies an available channel, and sends a message on the outbound control channel for all devices in the requested group to change to the available channel.

Reservation methods require a device to reserve a communication channel prior to transmission. Time is divided into superframes, each superframe is then further divided into a reservation period and a data-transmission period. During the reservation period devices communicate that they want to reserve a data slot; fixed reservation period slots need to be assigned for each device.

Polling is the most straightforward way to perform demand-based channel access. A controlling device in the network repetitively asks all network devices, one by one, if they need channel access. Devices that do not need channel access decline/ignore the request, while nodes that require channel access inform the controller of that fact. The controller then assigns channel access for the requesting device.

4.2.1.3 Contention access

In contention access protocols nodes contend for a shared channel among each other for channel access; devices that lose access to the channel waits for a back-off period, and then try again. Since frame collisions are not prohibited by contention-access protocols, a method for detecting collisions and recovering from them must be included in the MAC protocol.

ALOHA is the simplest form of random access for wireless digital communications. The first version, *Pure ALOHA*, involved sending a message immediately when generated, if a message collision occurs. The message is resent later using a backoff scheme. The backoff scheme significantly influences the efficiency of the protocol. *Slotted ALOHA* improved on the original by introducing discrete time slots and increased the maximum throughput by reducing collisions.

Carrier Sense Multiple Access (CSMA) is a probabilistic MAC protocol in which a node verifies the absence of other traffic before transmitting on a shared transmission medium. When a station has data to send, it first listens to the channel to see if anyone else is transmitting at that moment. If the channel is busy the station waits until it becomes idle. When the station detects an idle channel it transmits a frame. If a collision occurs the station waits a random amount of time and starts all over again. The protocol is called *1-persistent* because the station transmits with a probability of 1 when it finds the channel idle. *Nonpersistent CSMA* provides better channel utilisation by not continually sensing the channel when it is in use, however it provides longer delays. *P-persistent CSMA* stations transmit based on the probability p when it senses the channel as idle. *CSMA with Collision Avoidance (CSMA/CA)* attempts to avoid collisions by using a control message exchange to reserve the wireless channel before each data message transmission. This RTS/CTS channel allocation mechanism helps towards collision avoidance and combat the possible presence of hidden nodes. The RTS/CTS mechanism was originally described as part of the MACA protocol and MACAW [51] followed by adding link layer acknowledgements to create a fully handshaked protocol for reliable transfers.

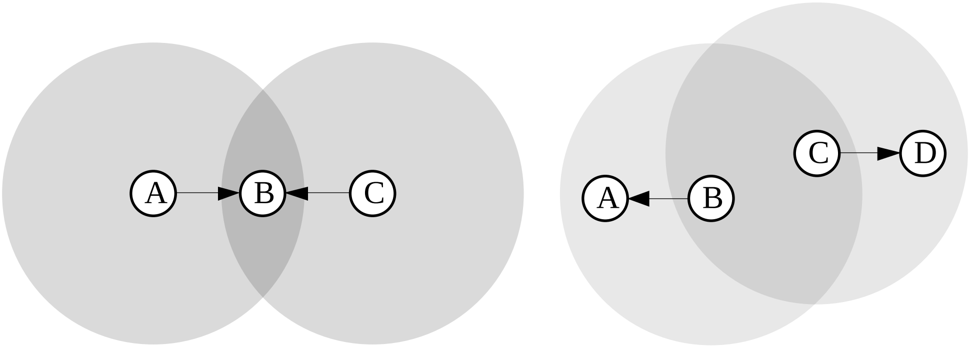
4.2.2 Fundamentals

This section presents some fundamental concepts related to MAC protocol design: the major sources of energy waste that should be addressed and the hidden- and exposed-terminal problem which expressed the need for RTS/CTS handshaking.

In the MAC layer there are several major sources of energy waste which should be minimised to achieve greater energy efficiency [2]: idle listening, overhearing, over-emitting, network signalling overhead, and collisions.

Idle listening occurs when the radio transceiver is on even though there is no data to transmit or receive. It has been studied that sending and receiving packets of various sizes indicate that the power consumed when the interface is on and idle is virtually identical to the cost of receiving packets [3]. *Overhearing* occurs when a node receives and decodes packets that are not destined to it. Because the wireless medium is a broadcast medium, any nodes within the transmission range will receive packets, even if it is not destined to them. *Over-emitting* occurs when the transmitter node transmits a packet while the receiver node is not ready to receive. Over-emitting exist within contention-access protocols such as ALOHA and CSMA. All protocols include some form of *network signalling overhead*. Per-packet overhead (packet headers/trailers) and control packets contribute to this. Protocol overhead is necessary to perform the required tasks, but should be kept to a minimum. *Collisions* occur when there are simultaneous transmissions from several nodes that are within the interference range of the receiver node. Since the collided packets must be retransmitted, it causes unnecessary energy waste.

Hidden terminal interference is caused by the simultaneous transmission of two node stations that are not within communication range of each other, but the receiver experiences a collision that cannot be detected by the transmitting nodes [52]. Figure 4.4(a) illustrates the hidden terminal problem in CSMA. Suppose that node A wants to transmit to node B. By only sensing the medium, node A will not be able to hear transmissions by node C, and will start transmitting, leading to collisions at node B (from node C). The hidden terminal problem is partially solved using RTS/CTS handshaking. RTS/CTS creates significant per-packet overhead.



(a) Hidden terminal scenario.

(b) Exposed terminal scenario.

FIGURE 4.4: Hidden- and exposed-terminal problem scenarios. Circles indicating communication range of each device.

The *exposed node* problem occurs when a node is prevented from sending packets to other nodes due to a neighbouring transmitter [52]. Consider Figure 4.4(b), where the two receivers (A and D) are out of range of each other, yet the two transmitters (B and C) are in range of each other. Here, if a transmission between A and B is taking place, node C is prevented from transmitting to D as it concludes, after carrier sensing, that it will interfere with the transmission by its neighbour node B. However, node D could still receive the transmission from C without interference because it is out of the range of B.

4.2.3 Contention-based protocols

In contention access protocols nodes compete for a shared channel while trying to avoid frame collision. CSMA is the most commonly used method, using channel reservation (RTS/CTS) and virtual carrier sensing (NAV) to reduce collisions and overhearing.

The *power aware medium-access protocol with signalling (PAMAS)* [53] protocol introduced NAV to improve the energy savings by turning nodes off for the duration of a packet transmission if they are not the intended destination of the packet, effectively reducing energy wastage because of overhearing. It uses RTS/CTS signalling over a separate signalling channel to reduce collisions. PAMAS lack features to reduce the energy wastage because of idle listening.

Sparse topology and energy management (STEM) [54] proposed using separate radios for data and wake-up in different channels. The data radio is by default in sleep mode and the wake-up radio periodically wakes itself up to perform preamble listening. To send data, the transmitter sends a preamble on the wake-up channel. STEM proposed to send successive small packets for a preamble instead of one long preamble to reduce transmission costs associated with a long preamble.

Sensor MAC (S-MAC) [44] provides a tuneable, non-adaptive, periodic active/sleep cycle for sensor nodes. S-MAC includes RTS/CTS signalling for collision avoidance and virtual carrier sensing (NAV) for overhearing avoidance. Duty cycled sleep scheduling reduce idle listening but this can cause tail-dropping in high-traffic scenarios where the available transmission time is too low. Another problem with S-MAC is that a single node can follow multiple schedules which severely impacts the average duty cycle. *Dynamic Sensor MAC (DS-MAC)* [55] is able to dynamically change the sleeping interval with fixed listen interval length and therefore the duty cycle of sensors is adjusted to adapt to traffic conditions. Therefore, DS-MAC alleviates the high latency problem presented in S-MAC when the traffic load is high, while still keeping the energy efficiency when the traffic load is low.

Timeout MAC (T-MAC) [9] includes all the features of S-MAC and adds an adaptive listening scheme which adapts the active portion of each cycle dynamically (in contrast to DS-MAC which changes the sleeping interval). As with S-MAC, collision avoidance is reduced using RTS/CTS signalling, overhearing is reduced using virtual carrier sensing (NAV), and idle listening is reduced by sleep scheduling. The T-MAC protocol improves on the sleep scheduling by dynamically adjusting the active portion to reduce idle listening even further under low traffic environments and allowing longer active periods during high traffic.

The *Berkley MAC (B-MAC)* protocol is highly reconfigurable and lightweight, features can be turned on/off or used in combination as necessary. The B-MAC protocol features asynchronous sleeping, low-power listening, clear channel assessments (CCA), and acknowledgements (ACK) for unicast traffic. Channel reservation using RTS/CTS signalling is not included.

Data-gathering MAC (D-MAC) [11] utilises a staggered schedule to exploit the tree-structure of source-to-sink communication and reduce reporting latency. In DMAC, medium reservation (RTS/CTS) was not included because it would add unnecessary overhead, this assumption was based on small packet sizes and bandwidth requirements. The lack of medium reservation does not allow for the scalability needed for next-generation WSNs. However, link layer automatic repeat request (ARQ) through ACK packets are necessary to recover lost packets due to harsh quality wireless channel and contention. Data-prediction is employed to solve problems where the aggregated data rate at nodes closer to the sink is larger than the basic duty cycle. Interference between nodes with different parents are almost guaranteed (no medium reservation) but D-MAC included features to reschedule lost packets.

The *mobility-aware MAC protocol for sensor networks (MS-MAC)* [56] uses changes in received signal level as an indication of mobility and, when necessary, triggers the mobility handling mechanism. During stationary operation MS-MAC works similar to S-MAC using periodic coordinated sleep/wake-up duty cycle. For a highly mobile scenario it switches to an operating mode similar to IEEE 802.11. The mobile aware mechanism forms an active zone around a mobile node, in which the synchronisation periods are more frequent. In the active zones nodes stay awake longer. This technique reduces the time a node would need to switch between virtual clusters.

WiseMAC [6] uses asynchronous sleeping based on the preamble sampling technique and non-persistent CSMA. It builds on the shortcomings of the preamble sampling technique: the long preamble which can cause throughput reduction and energy wastage for both sender and receiver. Nodes learn about the periodic sampling times of their neighbour nodes and use this information to reduce the wake-up preamble length which saves energy. The protocol also makes the necessary adjustments to cater for clock drift.

Preamble sampling is used to minimise idle listening, but no features were included to reduce collisions and overhearing.

4.2.4 Fixed-assignment protocols

In fixed-assignment protocols nodes are assigned unique time slots, frequency channels, or spreading codes for transmissions, eliminating collisions. This simplifies individual transmissions, but the bandwidth must be reserved prior to data transmissions increasing signalling traffic while trying to avoid frame collisions. The large nodal populations and dense deployments of WSNs make it difficult to provide fixed schedules that would contribute to very low bandwidth utilisation. Time synchronisation is necessary for fixed-assignment protocols that use global time for scheduling purposes (scheduling can be performed using timing-offsets which do not require time synchronisation).

Self-Organising Medium Access Control for Sensor Networks (SMACS) [57] is a distributed protocol based on a TDMA/FDMA hybrid scheme. Nodes discover their neighbours and establish transmission/reception schedules for communicating with them without the need for any local or global master nodes. Each node maintains a TDMA superframe, in which it schedules time slots to communicate with its known neighbours. The schedules are created using asynchronous scheduling communication methods. To address collisions FDMA uses different frequencies for each neighbour, which is chosen randomly from a large pool of frequencies when the links are established. The drawback of SMACS is the low bandwidth utilisation because it cannot reuse time slots and it does not provide sleep scheduling.

Node-Activation Multiple Access (NAMA) [58] uses a distributed election algorithm to achieve collision-free transmissions. For each TDMA time slot, NAMA selects one transmitter per two-hop neighbourhood and hence all the nodes in the one-hop neighbourhood of the transmitter are deemed to receive data in a collision-free manner. NAMA does not include features to reduce idle listening and bandwidth utilisation is low in dense deployments; TDMA time slots are not recycled when nodes have no data to send.

Traffic-adaptive medium access protocol (TRAMA) [59] reduces energy consumption by ensuring that unicast and broadcast transmissions incur no collisions and by allowing nodes to assume a low-power, idle state whenever they are not transmitting or receiving. This is established by exchanging the two-hop neighbourhood information. TRAMA uses a distributed TDMA election scheme based on information about traffic at each node to determine which node can transmit at a particular time slot. TRAMA enhances NAMA by using traffic information to avoid assigning time slots to nodes with no traffic to send, and allows nodes to determine when they can switch off and not listen to the channel. The drawback of TRAMA is that it requires active signalling in order to ensure synchronisation.

4.2.5 IEEE 802.15.4 MAC

The IEEE 802.15.4 MAC protocol provides a power-saving mechanism for a master-slave star topology. The master broadcasts synchronisation information using a periodic beacon. The beacon describes the superframe structure and indicates when slave nodes can sleep. Slave nodes need to awake simultaneously to listen to the beacon. The beacon contains information to which slaves should stay active: if the beacon indicates a pending packet. A slave also stays active if it has a pending packet for the master. Otherwise, the slave goes to sleep until the next beacon. This beacon listening scheme is not very efficient in two aspects. Firstly, the beacon is a complete MAC frame that increases the protocol overhead. Secondly, the beacon-enabled architecture only fits a simple star topology which is not efficient in dense deployments.

The IEEE 802.15.4 standard [26] is based on the OSI model and defines the physical and data link layers. The standard is designed for the low bit-rate wireless personal area networks (LR-WPAN). Three different physical layers have been defined, all operating in licence free ISM frequency bands: 250 Kbps rate (16 available channels) in the 2.4 GHz band, 40 Kbps rate (10 available channels) in the 915 MHz band, and 20 Kbps rate (single channel) in the 868 MHz band.

The LR-WPAN standard allows the optional use of a superframe structure (beacon-mode). The format of the superframe is defined by the coordinator. The superframe is bounded by network beacons sent by the coordinator and is divided into 16 equally sized slots, as shown in Figure 4.5. The beacon frame is transmitted in the first slot of each superframe. If the coordinator does not wish to use a superframe structure, it may turn off the beacon transmissions. The beacons are used to synchronise the attached devices, to identify the PAN, and describe the structure of the superframes. Any device wishing to communicate during the contention access period (CAP) between two beacons shall compete with other devices using a slotted CSMA/CA mechanism. All transmissions shall be completed by the time of the next network beacon.

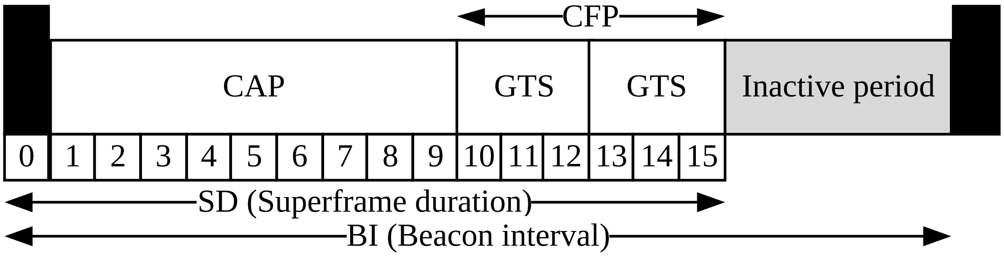


FIGURE 4.5: Superframe structure.

The superframe can have an active and an inactive portion. During the inactive portion, the coordinator shall not interact with its PAN and may enter a low-power mode. Equation 4.2 calculates the superframe duration (SD) and Equation 4.1 calculates the beacon interval (BI). These values are determined by the beacon order (BO) and superframe order (SO). When $BO = SO$ there is no inactive period; as the BO increases, the duty cycle decreases.

$$BI = aBaseSuperframeDuration \cdot 2^{BO} \quad where \quad 0 \leq BO \leq 14 \tag{4.1}$$

$$SD = aBaseSuperframeDuration \cdot 2^{SO} \quad where \quad 0 \leq SO \leq BO \leq 14 \tag{4.2}$$

For low-latency applications or applications requiring specific data bandwidth, the PAN coordinator may dedicate portions of the active superframe to that application. These portions are called guaranteed time slots (GTSs). The GTSs form the contention-free period (CFP) which always appears at the end of the active superframe, starting at a slot boundary immediately following the CAP, as shown in Figure 4.5. The PAN coordinator may allocate up to seven of these GTSs, and a GTS may occupy more than one slot period. A sufficient portion of the CAP shall remain for contention-based access for other networked devices or new devices wishing to join the network. All contention-based transmissions shall be completed before the CFP begins. Each device transmitting in a GTS shall ensure that its transmission is complete before the time of the next GTS or the end of the CFP.

4.3 NETWORK LAYER

Sensor nodes are scattered densely in a field to perform some kind of sensing function. The limited communication range of the sensor nodes prevents direct communication between all sensor nodes and the data sink. Multi-hop wireless routing protocols are required to create data paths between sensor nodes and the data sink.

The most commonly used network configuration in WSNs is the spanning tree structure. The spanning tree mimics the data gathering path and is rooted at the data sink to minimise uplink and download communication costs [60]. Forwarding messages in a spanning tree has the advantage of locating the lowest cost path between each of the nodes and the data sink, which requires minimum cost forwarding and source routing to perform effectively [61]. The tree-style nature of the spanning tree structure lends itself to the use of data aggregation and compression to reduce the data traffic traveling upstream.

4.3.1 Classification of routing methods

WSN routing protocol performance is closely related to the architectural model. There are several factors that influence the design considerations for WSN routing protocols.

These factors include: node mobility, node deployment, energy consumption, data delivery methods, heterogeneity and data aggregation [62]. Figure 4.6 shows the different routing protocol categories.

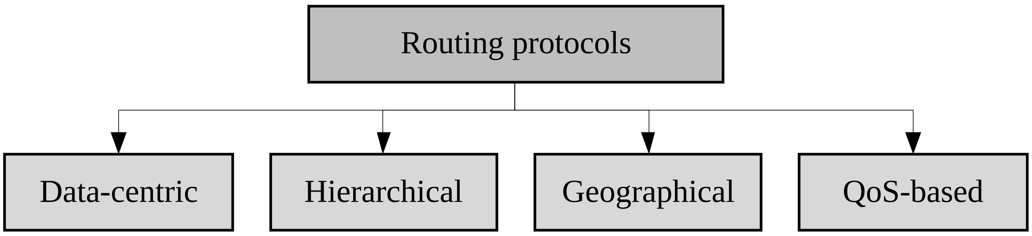


FIGURE 4.6: Routing protocol taxonomy.

Routing paths can be established in one of two ways: reactive or proactive. Reactive protocols compute routes only when needed and the assumption is that very little data traffic will be generated. Proactive protocols compute all routes before they are needed and topology changes propagate across the network as they happen. Since a WSN consists of many nodes and is data-centric, distance vector algorithms using the Bellman-Ford algorithm are more suitable than link-state algorithms which require global topology information.

A fundamental innovation in the area of WSNs has been the concept of data-centric routing. The idea is that communication is based directly on application-specific data content instead of the general address-based communication networks. In data-centric routing the sink sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data.

The main aim of hierarchical routing is to efficiently maintain the energy consumption of sensor nodes by involving them in multi-hop communication within a particular cluster and by performing data aggregation and fusion in order to decrease the number of transmitted messages to the sink. A hierarchical cluster scheme allows a large number of nodes to be inactive, while a small percentage of nodes stay awake to ensure network connectivity.

Geographical routing protocols utilise the location of nodes to perform efficient and scalable routing. Physical distance is closely related to the energy consumed to forward data. Based on the location information of sensors, protocols can provide short- or long range communication paths, depending on whether the distance between consecutive forwarders should be minimised or maximised. Closer distances tend to provide better signals and less power can be used for the transmission. Nodes which maximises distance can reach a data sink in less hops but the reliability of the channel can be affected.

Sensing applications may have different requirements which can be expressed in terms of some Quality of Service (QoS) metrics, such as latency, reliability, bandwidth, and fault tolerance. The finite energy supply in WSN nodes introduced new unique routing metrics and protocols which rely on the battery status. These methods aim to increase the network lifetime by exploiting the distributed battery status of nodes.

4.3.2 Data-centric

The simplest data-centric routing algorithm is *flooding*. Whenever a node receives a packet, it broadcasts this packet to all neighbours. Flooding is a very low-complexity reactive protocol and does not require any network topology control. However, flooding is subject to implosion, overlapping and resource blindness. Implosion occurs when a node receives duplicate packets from multiple sources; implosion is unsuitable for large multi-hop networks. Overlapping refers to the concept where spatial correlation of events are not taken into consideration and the same event can be reported by multiple nodes. Flooding does not take resource metrics into account and is thus not energy-aware.

Sensor Protocols for Information via Negotiation (SPIN) [63] was designed to improve classic flooding protocols and overcome the associated problems. The SPIN protocol is resource aware and resource adaptive. SPIN enable sensors to negotiate with each other before any data dissemination can occur, in order to avoid injecting redundant information. Nodes use meta-data to describe the data that they have available. SPIN provides local resource adaption: each sensor is resource aware and can decide not to participate in data forwarding in order to extend its operating lifetime.

Directed diffusion [64] is data-centric because communication is for named data. All nodes in a directed diffusion network are application aware. Directed diffusion achieves energy savings by selecting empirically good paths and by caching and processing data in-network. Directed diffusion consists of several elements: interests, data messages, gradients, and reinforcements. An interest message is a query or an interrogation which specifies the user request. A data message is the collected or processed information of a physical sensed event. An interest is disseminated throughout the sensor network and gradients are set up in the opposite direction. A single data delivery path is reinforced for communications.

Rumour routing [65] is intended to fill the gap between query flooding and event flooding. The idea is to route the queries to the nodes that have observed a particular event rather than flooding the entire network to retrieve information about the occurring events. In order to flood events through the network, rumour routing employs long-lived packets, called agents. When a node detects an event, it adds such event to its local table and generates an agent. Agents travel the network in order to propagate information about local events to distant nodes. When a node generates a query for an event, the nodes that know the route, can respond to the query by referring to its event table. Hence, the cost of flooding the whole network is avoided. Rumour routing maintains only one path between source and destination as opposed to directed diffusion where data can be sent through multiple paths at low rates.

4.3.3 Hierarchical

Low Energy Adaptive Clustering Hierarchy (LEACH) [66] is a self-organising, adaptive, clustering protocol that uses randomisation to distribute the energy load evenly among the sensors in the network. In LEACH nodes organise themselves into local clusters, with one node acting as the local base station or cluster-head. LEACH includes randomised rotation of the high-energy cluster-head position such that it rotates among the various sensors in order not to drain the battery of a single sensor. Once the clusters are formed, the cluster heads broadcast a TDMA schedule giving the order in which the cluster members can transmit their data.

Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [67] is a greedy chain protocol that is an improvement of the LEACH protocol. The main concept of PEGASIS is for nodes to receive and transmit to close proximity neighbours and take turns being the cluster-head for transmission towards the data sink. This technique promotes evenly distributed load amongst nodes. The problem with PEGASIS is that it assumes that each sensor is in the communication range of all nodes and does not represent a true multi-hop network.

Threshold sensitive Energy Efficient sensor Network protocol (TEEN) [68] is targeted at reactive networks and is best suited for time critical applications such as intrusion detection, explosion detection, etc. The sensor network architecture is based on a hierarchical grouping where close proximity nodes form clusters and this process continues until the base station (data sink) is reached. The main drawback of TEEN is that when the thresholds are not reached, nodes will never communicate and the users will not get any data from the network nor will they know if any nodes died. TEEN is not well suited for applications where the user needs to get data on a regular basis. The TEEN protocol does not mention how to mitigate problems such as collisions, idle listening and overhearing. *Adaptive TEEN (APTEEN)* [69] is an extension of TEEN which aims at adding the ability of capturing periodic data collections and being reactive to time-critical events. APTEEN adds collision avoidance using a TDMA scheme.

4.3.4 Geographical

Geographical energy-aware routing (GEAR) [70] is proposed for routing queries to target regions in a sensor field. The idea behind GEAR is to restrict the number of interests in Directed Diffusion by only considering a certain region rather than sending interests to the whole network. GEAR uses energy aware and geographically informed neighbour selection heuristics to route a packet towards the target region. Within a region, it uses a recursive geographic forwarding technique to disseminate the packet. When a sensor needs to forward a packet, it selects a neighbour that is closer to the destination region than itself. When the packet reaches the destination region, the region is split into sub-regions, and a packet is forwarded to each sub-region. This split-forward sequence is repeated until the region contains only one node.

Geographical adaptive fidelity (GAF) [71] conserves energy by identifying nodes that are equivalent from a routing perspective and then turning off unnecessary nodes, keeping a constant level of routing fidelity (network connectivity). GAF divides the operating field into grid squares and each node identifies with a particular grid in which it resides. The node with the highest residual energy level acts as the active node in each grid. Each node follows a state transition with three states: discovery, sleep, and active. In the sleep state, the node turns off its radio for energy savings. During the discovery state a sensor exchanges information with other nodes in its grid. During the active state a node acts as the forwarding node for the grid.

Minimum Energy Communication Network (MECN) [72] describes a distributed protocol to find the minimum power topology for a stationary ad hoc network. The motivation of MECN is based on the key premise that maximising the total battery lifetime of a network requires minimising the energy consumption of the entire network. MECN is a self-reconfiguring protocol that maintains network connectivity in spite of sensor mobility. It computes an optimal spanning tree rooted at the sink, referred to as the minimum power topology, which contains only the minimum power paths from each sensor to the sink. In MECN, it is assumed that every node can transmit to every other node, which is not possible every time. In *Small Minimum Energy Communication Network (SMECN)* [73] possible obstacles between any pair of nodes are considered, avoiding bad communication links. However, in SMECN it is still assumed that every node can transmit to every other node, as in the case of MECN.

4.3.5 Quality of Service

The objective of the *Sequential Assignment Routing (SAR)* [57] algorithm is to minimise the average weighted Quality of Service (QoS) metric throughout the lifetime of the network. As each path is used over time, the available energy resource will change. There are also possible changes in the QoS on each path. These changes will be accounted for by a periodic metric update triggered from the sink node. The SAR protocol creates trees rooted at one-hop neighbours of the sink by evaluating the QoS metric (available energy resources on a path) and priority level of each packet. Each link contributes an energy cost and delay, the resistance to packet flow can be captured in an additive metric for any given path. By using created trees, multiple paths from sink to sensors are formed. One of these paths is selected according to the energy resources and QoS on the path.

SPEED [74] is specifically tailored to be a stateless localised algorithm with minimal control overhead. The protocol requires each node to maintain information about its neighbours and uses geographic forwarding to find the paths. In addition, SPEED strives to ensure a certain speed for each packet in the network so that each application can estimate the end-to-end delay for the packets by dividing the distance to the sink by the speed of the packet before making the admission decision. SPEED uses a backpressure re-routing scheme to re-route packets around large-delay links with minimum control overhead.

Collection Tree Protocol (CTP) [75, 76] is a tree-based collection protocol which forms a spanning tree structure. Some number of nodes in a network advertise themselves as tree roots (data sinks). Nodes form a set of routing trees to these roots. CTP is address-free in that a node does not send a packet to a particular root; instead, it implicitly chooses a root by choosing a next hop. Nodes generate routes to roots using a routing gradient; CTP uses expected transmissions (ETX), a measure of link quality, as its routing gradient. However, relying on existing shortest path or minimum-weight spanning tree algorithms may not be adequate since the resultant spanning tree and routing paths have to negotiate around the unique WSN operational constraints such as large nodal populations and sleep scheduling.

The *Minimum Cost Path Forwarding* protocol combines link characteristics such as delay, throughput, energy consumption, and geographical distance to establish routes between nodes in the network [49]. The protocol assigns a cost function to each link. Cost field establishment determines the minimum cost between any node and the sink. The data sink broadcasts an advertisement (ADV) message with an initial cost of 0. The ADV message is then forwarded after adding the link cost.

In Table 4.3 the most common metrics used in Wireless Mesh Networks are displayed. Additional metrics for WSNs have been proposed based on energy efficiency requirements [2]: Minimise energy per packet, Maximum Total Available Battery Capacity, Minimum Battery Cost Routing (MBCR), Min-Max Battery Cost Routing (MMBCR), Conditional Max-Min Battery Capacity Routing (CMMBCR), and Minimum variance in power levels.

TABLE 4.3: Minimum Cost Path Forwarding metrics [77].

Metric	**Name**	**Quality aware**	**Data rate**	**Packet size**	**Intra-flow interference**	**Inter-flow interference**	**Medium instability**
HOP	Hop count	X	X	X	X	X	X
ETX	Expected transmission count	√	X	X	X	X	X
ML	Minimum loss	√	X	X	X	X	X
ETT	Expected Transmission Time	√	√	√	X	X	X
WCETT	Weighted Cumulative Expected Transmission Time	√	√	√	√	X	X
MIC	Metric of Interference and Channel Switching	√	√	√	√	√	X
ENT	Effective number of transmissions	√	√	√	X	X	√
iAWARE	Interference Aware	√	√	√	√	√	√

4.4 SUMMARY

This chapter introduced a comprehensive study on the communication protocols and techniques used in WSNs. The physical layer is responsible for the conversion of the bit-stream into signals that are best suited for communications across the wireless channel.

The physical layer performs frequency selection, carrier frequency generation, signal detection, modulation, and data encryption.

The data link layer provides addressing and channel access control mechanisms that make it possible for several network nodes to communicate reliably within a point-to-point or point-to-multipoint network. A fundamental task of the wireless MAC protocol is to avoid collisions so that two interfering nodes do not transmit at the same time. In the MAC layer there are several major sources of energy waste which should be minimised to achieve greater energy efficiency: idle listening, overhearing, over-emitting, network signalling overhead, and collisions.

The network layer creates the necessary infrastructure to support multi-hop communication which enables nodes to send their data to a nearby data sink. The different WSN challenges lead to the creation of unique WSN routing protocols which can be categorised into data-centric, hierarchical, geographical, or QoS-based protocols.

CHAPTER 5

PROTOCOL DESIGN

Objectives of this Chapter

This chapter provides the proposed protocol stack for wireless sensor networks to enable low complexity, ultra-low energy consumption, and low data rate wireless connectivity amongst inexpensive devices. The protocol stack is aimed at large nodal populations, densely deployed, with periodic sampling applications. It uses the IEEE 802.15.4 PHY standard in the 2.4 GHz frequency band. A novel hybrid data-link/network cross-layer solution is proposed: a global sleep schedule, geographical data gathering tree, TDMA-slotted architecture, CSMA/CA, CCA with a randomised contention window, adaptive listening using a conservative timeout activation mechanism, virtual carrier sensing, clock drift compensation, and error control.

The objective of the proposed design is to achieve an ultra-low duty cycle sleep scheduling protocol stack for wireless sensor networks. The protocol tries to reduce energy consumption from all the identified sources that cause energy waste, i.e., idle listening, overhearing, over-emitting, collisions, and network control overhead. The protocol optimisation goals are described as follows:

Extended lifetime The protocol should strive to extend network lifetime. An ultra-low (below 1 %) duty cycle sleep schedule should be achieved.

Responsiveness The protocol should ensure that traffic arrives at the sink in a timely manner; data should arrive at the sink before the originating node generates another measurement.

Robustness The protocol should ensure that the global performance of the system is not sensitive to individual device failures.

Scalability The protocol must be scalable and function efficiently for networks of any size.

Self-configuration The protocol must provide autonomous operation of the network. The design should have the ability for network nodes to detect the presence of other nodes and to organise into a structured, functioning network without human intervention.

5.1 ASSUMPTIONS

During the design phase certain assumptions were made regarding the application and deployment architectures for the proposed protocol. Localisation and time synchronisation were required for the implementation of the protocol, but their implementations were beyond the scope of the dissertation (the simulator allowed direct access to these values and no implementations were needed).

Application A periodic sampling application with a low sampling frequency; a 5 min (300 s) sampling period was selected for all simulations and design decisions.

Mobility A stationary network was assumed; nodes do not have the ability to adjust their location (mobilisation).

Low Traffic A very low data traffic rate was assumed. The assumption was based on the periodic sampling rate and the network size.

Localisation The implementation assumed fine-grained localisation: all nodes know their own geographical coordinates. The network overhead for the localisation was not been included, but it is assumed to be very low in stationary networks.

Homogeneity It was assumed that all nodes have exactly the same capabilities and are homogeneous in nature.

Time synchronisation Coarse time synchronisation was assumed for all nodes. The overhead for time synchronisation was not included.

Clock drift It was assumed that clock drift is present in all nodes, with a maximum deviation of 30 ppm.

Deployment Large nodal populations, dense deployments, and randomised positioning were assumed.

Data gathering A unidirectional traffic path was assumed. A source-to-sink data reporting tree structure approach was followed.

5.2 PROTOCOL DESCRIPTION

5.2.1 Functional overview

The characteristics of the proposed protocol are:

Slotted architecture Each time frame is divided into multiple time slots, similar to a TDMA scheme. Each slot has a fixed duration of 10 s. A slotted architecture provides reduced interference by dividing the nodes within the network into different slots, reducing the contention per time slot. Each time slot represents a level in the data gathering tree.

Global sleep schedule Idle listening is reduced by creating a global sleep schedule. The sleep schedule correlates to the slotted architecture. Nodes awake in their own slot to transmit a synchronisation beacon and listen for data. A node will awake in its parent's slot to listen for synchronisation beacons and transfer any buffered data. To increase the propagation speed of network changes nodes will awake in all slots to listen for synchronisation beacons and update their routing tables.

CSMA/CA A contention-access protocol allows nodes to share the wireless transmission medium. Basic collision avoidance is provided by using Clear Channel Assessment (CCA); the CCA algorithm includes a randomised contention window period. A channel is assessed to be clear after two consecutive clear carrier sensing operations.

Medium reservation The RTS/CTS medium reservation method provides collision avoidance.

Reliability A fully handshaked protocol (RTS/CTS and data-link layer acknowledgements) is used for unicast data traffic to provide reliability.

Virtual carrier sensing A network allocation vector (NAV) field is included in RTS and CTS packets which allow nodes to sleep during transmission not intended for them. Virtual carrier sensing is performed to reduce overhearing.

Adaptive listening A timeout mechanism is utilised to adaptively shape the synchronisation beacon exchange duration and the data exchange period based on the traffic patterns and network requirements. A conservative activation-timeout method is used which increases reliability.

Data aggregation Data aggregation is performed during each slot to reduce network overhead. A node will stay awake in its own slot and gather data packets. In the next slot (parent slot), it will send the aggregated data packets.

Geographical routing The geographical location of nodes are used to create a routing table based on the distance between nodes. The routing algorithm minimises the distance between communicating nodes while ensuring that it moves closer towards the sink at each stage. A Bellman-Ford adaptive distance-based metric is used. Split horizon was implemented to eliminate routing loops.

Periodic sampling Each node generates a sample and transmits it towards the data sink every 5 min (300 s). A randomised jitter of 30 s was introduced to alleviate communication bursts and transmission deadlocks.

The proposed protocol uses a hybrid cross-layer data-link/network layer solution, as shown in Figure 5.1. In the OSI model, each layer operates separately and only provide fixed encapsulation interfaces to allow for modularity in design. The data link and network layer is tightly integrated: the geographical routing information is used by the MAC layer to set up the slotted architecture.

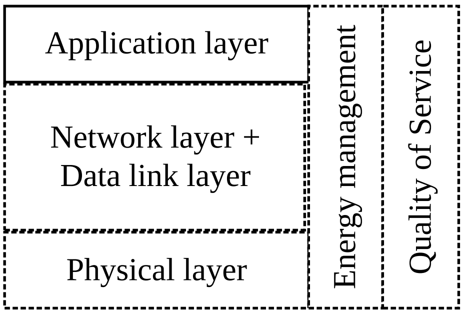
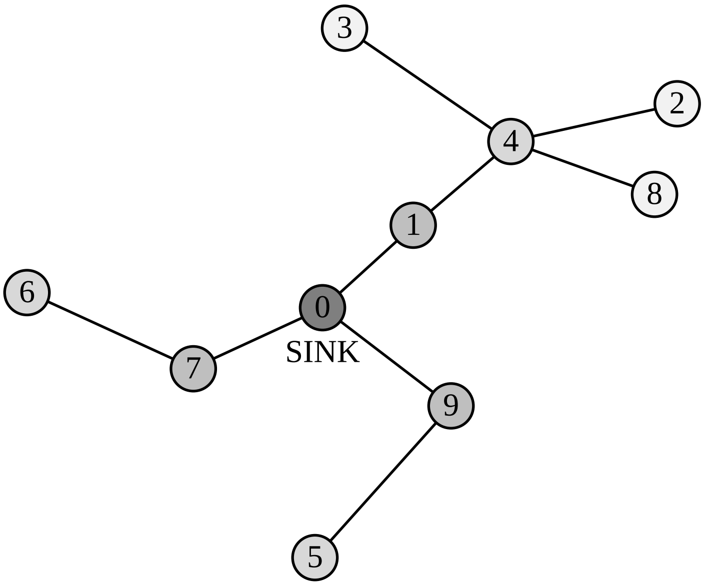


FIGURE 5.1: Hybrid protocol stack.

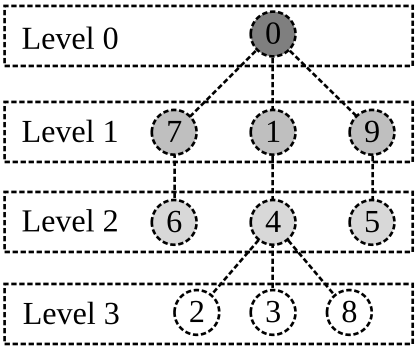
5.2.2 Architecture

Figure 5.2(a) shows the physical layout of a 10 node WSN and the corresponding routed forwarding links. Each line indicates a communication link between two nodes. Figure 5.2(b) illustrates the logical reporting tree structure for the network provided in Figure 5.2(a).

Node 0 is the data sink and acts as the gateway between the WSN and the transit network. Nodes 1, 7, and 9 are close to the data sink and falls into level 1 of the data reporting tree. Nodes 6, 4, and 5 are divided into level 2 and nodes 2, 3, and 8 fall into level 3. Nodes within each level is assigned a time slot and use CSMA/CA to access the channel. The maximum network depth of the network is 3, and data is aggregated from the lower levels up to the data sink (level 0).



(a) Physical layout and data reporting tree. Number of nodes = 10



(b) Logical data reporting tree. Number of nodes = 10

FIGURE 5.2: The architecture.

5.2.3 Network topologies

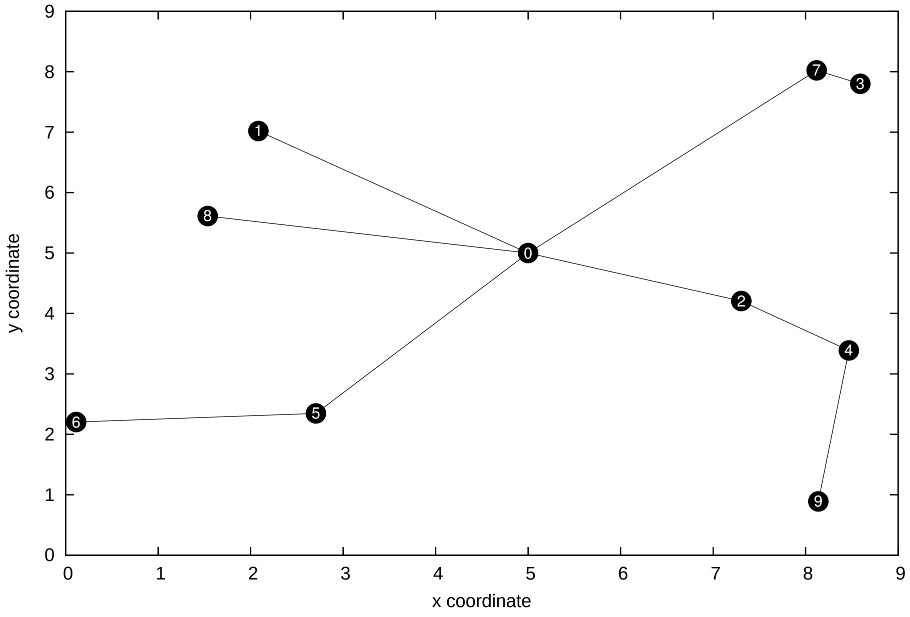
The network topology is dependent on the geographical layout of the network. Two different topological scenarios are evaluated: a network with 10 nodes and a network with 100 nodes. For each of these scenarios, two random topologies are assessed. These topologies are evaluated by investigating the layout of the data reporting tree structure. Data is generated by nodes and forwarded towards the data sink located in the middle of the layout.

In Figure 5.3(a) the maximum network depth is 3, formed by: $0 \leftarrow 2 \leftarrow 4 \leftarrow 9$. Thus, any data that node 9 has to transmit will be unicast to node 4. Node 4 will aggregate all the received data and will unicast the data towards node 2, where node 2 will do the same and send the aggregated data to the data sink. In this topology the data sink is in level 0, five nodes are divided into level 1, three nodes are allocated to level 2 and a single node resides in level 3.

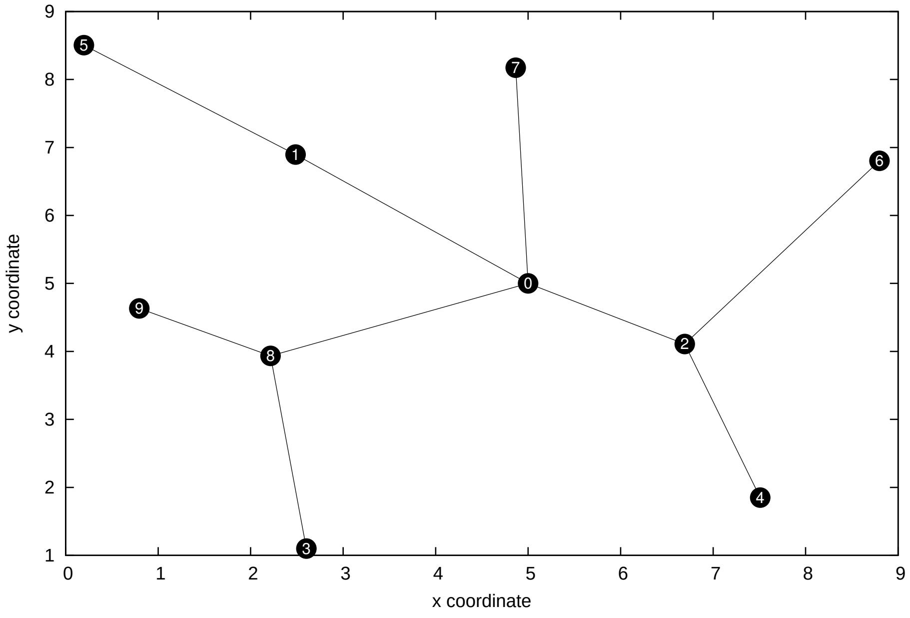
In Figure 5.3(b) the maximum network depth is 2. Multiple nodes have a level of 2, these include: $0 \leftarrow 1 \leftarrow 5$, $0 \leftarrow 2 \leftarrow 6$, $0 \leftarrow 2 \leftarrow 4$, $0 \leftarrow 8 \leftarrow 3$ and $0 \leftarrow 8 \leftarrow 9$. In this topology the data sink is in level 0, four nodes are divided into level 1, and five nodes are allocated to level 2.

In Figure 5.4(a) the maximum network depth is 11, in a network with 100 nodes. The maximum depth link-chain is formed by: $0 \leftarrow 96 \leftarrow 27 \leftarrow 81 \leftarrow 44 \leftarrow 87 \leftarrow 7 \leftarrow 3 \leftarrow 14 \leftarrow 41 \leftarrow 63 \leftarrow 37$. Figure 5.4(b) has a maximum network depth of 13 (also containing 100 nodes), which is formed by: $0 \leftarrow 81 \leftarrow 93 \leftarrow 63 \leftarrow 72 \leftarrow 54 \leftarrow 70 \leftarrow 46 \leftarrow 47 \leftarrow 91 \leftarrow 56 \leftarrow 35 \leftarrow 18 \leftarrow 19$.

From these examples it is clear that data aggregation will play an important role to reduce network overhead in large network topologies. These examples were extracted after network convergence and show the geographical routing protocol without routing loops.

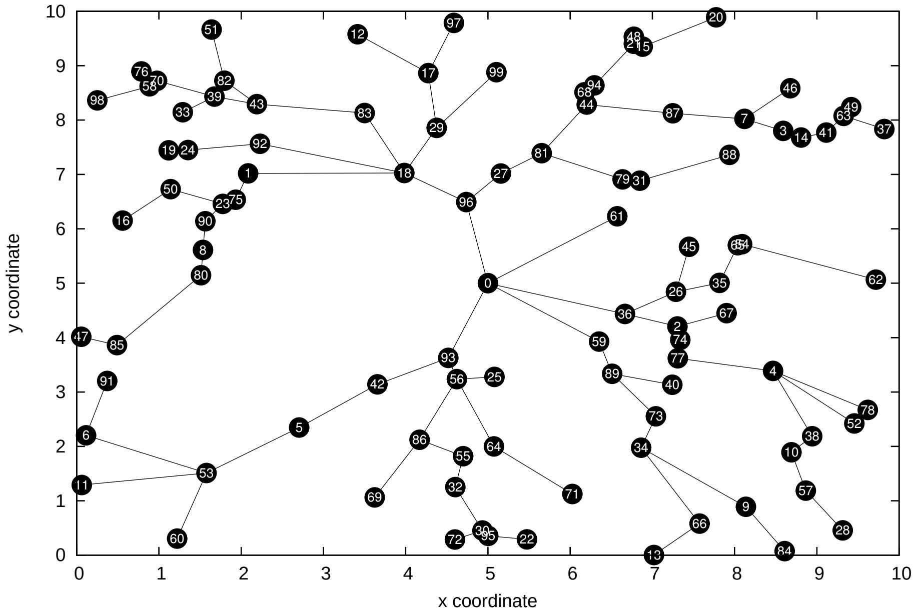


(a) Random number generator seed number = 2.

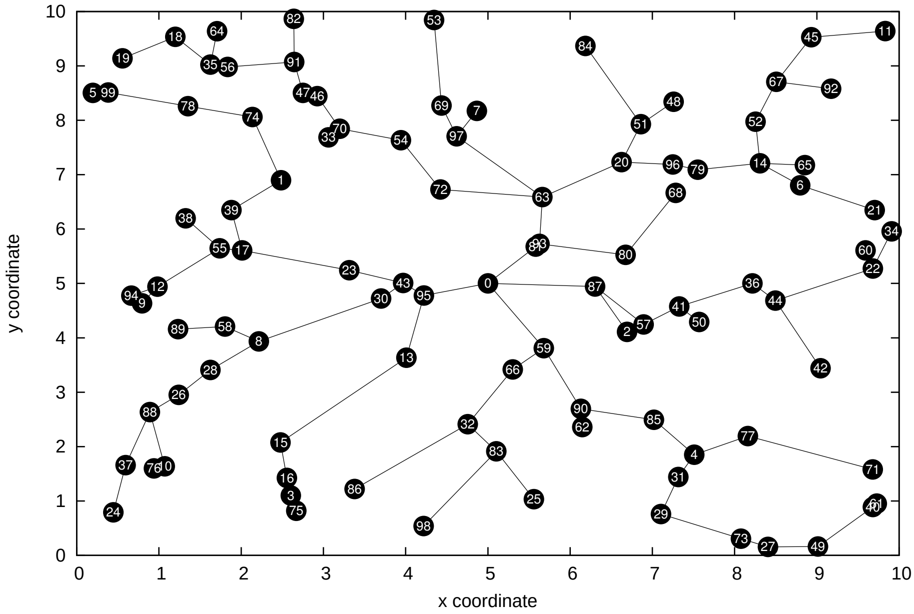


(b) Random number generator seed number = 3.

FIGURE 5.3: Physical layout and reporting structure. Number of nodes = 10.



(a) Random number generator seed number = 2.



(b) Random number generator seed number = 3.

FIGURE 5.4: Physical layout and reporting structure. Number of nodes = 100.

5.3 PHY SPECIFICATION

The proposed protocol uses the IEEE 802.15.4 PHY standard operating in the 2.4 GHz frequency band. This allows for a raw data rate of 250 Kbps. Figure 5.5 shows the PHY layer protocol data unit packet structure which adds a 6 byte header to all communication packets it receives from the data link layer.

Preamble	SFD	PHR
4 bytes	1 byte	1 byte

FIGURE 5.5: PHY packet.

The preamble field is used by the transceiver to obtain chip and symbol synchronisation of an incoming message. The start-of-frame delimiter (SFD) is an 8 bit field indicating the end of the synchronisation (preamble) field and the start of the packet data. The PHY header (PHR) includes the frame length field which specifies the total number of octets contained in the encapsulated data from the data link layer.

5.4 MAC SPECIFICATION

5.4.1 State diagrams

5.4.1.1 MAC states

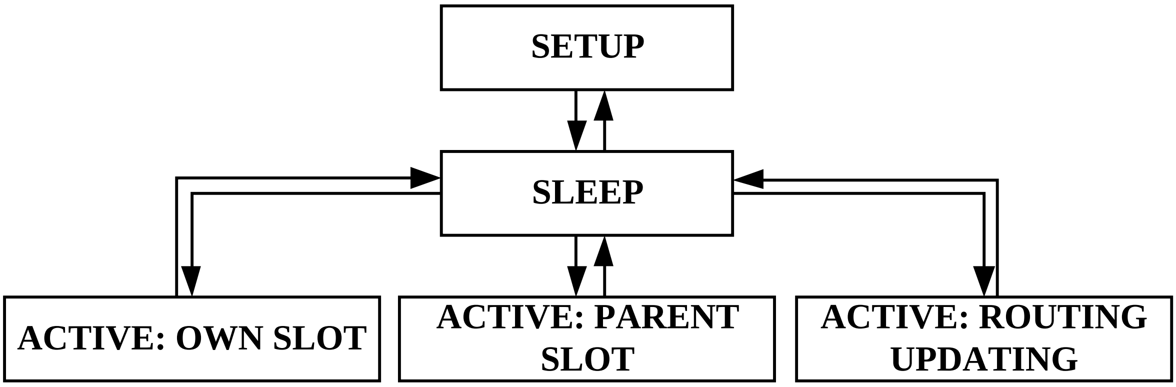


FIGURE 5.6: MAC state transitions.

Figure 5.6 shows the five different MAC states: SLEEP, SETUP, ACTIVE_OWN_SLOT, ACTIVE_PARENT_SLOT and ACTIVE_ROUTING_UPDATE. In the SLEEP state, a node will enter sleep mode to preserve energy; the SLEEP state is the inactive portion of the sleep schedule. The SETUP state is used during network convergence. The three different ACTIVE states are used to indicate the active periods of each frame, based on the ACTIVE state each node will operate differently.

In the SETUP state a node listens (radio in RX state) for any synchronisation beacon messages. The active listening period continues for 100 ms or until a synchronisation beacon is overheard. If no synchronisation beacon is found, a node will sleep for 10 s. This hard-limits the duty cycle during network convergence to 1%, eliminating the scenario where a node would actively search for a sleep schedule while no sink is present in the network. This can lead to nodes quickly draining all energy if no sleep schedule is present in the network convergence phase.

The ACTIVE_OWN_SLOT state indicates the slot in which a node transmits its own sleep scheduling using a SYNC packet, and listens during the DATA period for data from nodes in the level below. A node in the ACTIVE_PARENT_SLOT state will listen for SYNC packets during its parent's slot and will transmit buffered data during the DATA portion of the slot. In the ACTIVE_ROUTING_UPDATE state, a node will only listen during the SYNC period to update its routing table entries - it will not transmit a SYNC packet or listen during the DATA period.

5.4.1.2 Communication states

The radio state can be in either receive (RX), transmit (TX) or sleep mode. The radio switches to sleep mode during the SLEEP MAC state. The radio switches to TX state to transmit a packet, which could be a synchronisation beacon (SYNC), request-to-send (RTS), clear-to-send (CTS), DATA or acknowledgement (ACK) packet, otherwise the radio is assumed to be in RX state.

The unicasting TX state transition diagram is shown in Figure 5.7. When a node wants to transmit a unicast packet it transitions from the ACTIVE state to the clear channel assessment (CCA) state. During the CCA, the node performs carrier sensing and makes sure that the channel is clear. The RTS/CTS process starts by transmitting a RTS packet. The node then waits for the destination node to return a CTS packet. If a CTS packet is not received the session times out and returns back to the ACTIVE state; a node will retry medium reservation 10 times before waiting until the next frame. If the CTS is received, it will immediately transmit the DATA without performing another CCA (the channel was reserved during the RTS/CTS handshaking). A node then waits for the acknowledgement to finish the session. If the ACK is not received, the session times out and returns to the ACTIVE state; a node will retry transmission 10 times before waiting until the next frame.

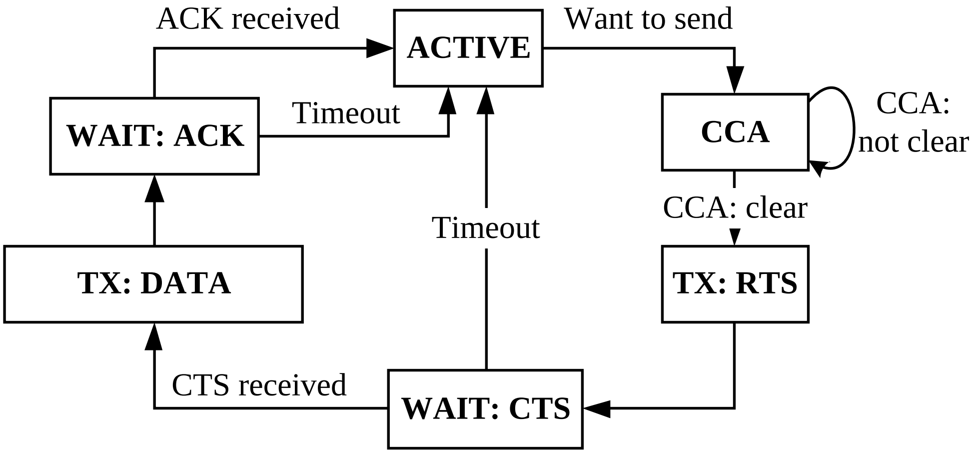


FIGURE 5.7: Unicast TX.

The broadcasting TX state transition is a much simpler process, see Figure 5.8. Broadcasting does not require link layer reliability and therefore the fully handshaked process is removed. When a node wants to send a broadcast packet it performs the CCA and then transmits the packet when the channel is clear.

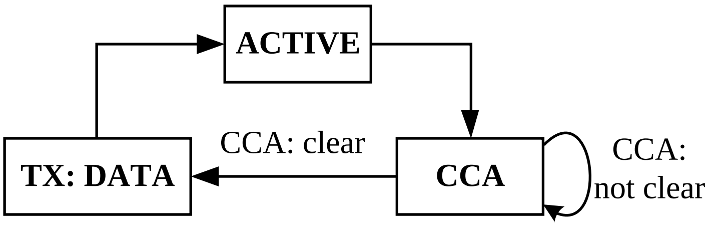


FIGURE 5.8: Broadcast TX.

Figure 5.9 shows the unicast RX transition states. When the radio of a node is in the ACTIVE RX mode, it can receive three different types of packets: RTS, DATA or overhearing of RTS/CTS packets. When a node receives a RTS it will reply with a CTS without performing a CCA (the RTS reserved the channel). If it receives a DATA packet, it will immediately reply with an ACK packet. Overhearing avoidance is performed using virtual carrier sensing using the NAV field within RTS/CTS packets. A node that overhears a unicast packet will look at the NAV field and sleep for the duration of the estimated communication indicated in the RTS/CTS packet.

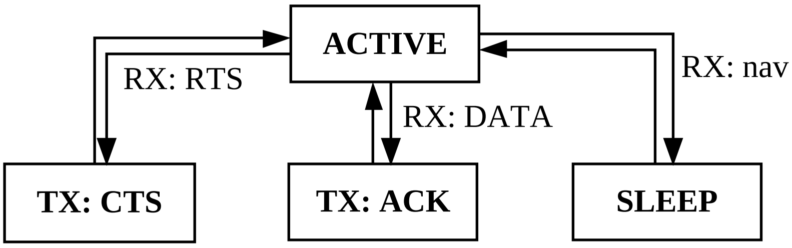


FIGURE 5.9: Unicast RX.

5.4.2 Slotted architecture

One of the unique characteristics of the proposed protocol is the slotted architecture, as shown in Figure 5.10. Each frame is divided into multiple slots. Each slot duration is 10 s long. To achieve an ultra-low (below 1 %) duty cycle sleep schedule, the active portion of each frame should be less than 100 ms. The ultra-low duty cycle sleep schedule will reduce energy wastage because of idle listening and prolong network lifetime.

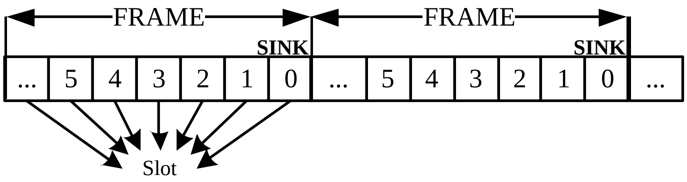


FIGURE 5.10: Slotted architecture.

A slotted architecture helps to reduce interference by dividing the nodes within the network into different levels. Nodes are spread out based on their geographical location and level in the data reporting tree. This technique will generate multiple geographical paths which help to alleviate data congestion.

A sink node (always allocated to slot 0) is in charge of the frame structure. The initial network depth of the network is 0; only level 0 is present. When a node wishes to join the network it will inspect a synchronisation beacon and find out the maximum level of the sleep schedule. If it is lower than the required level the node will flood a REGISTER packet in every time slot, requesting to increase the maximum depth of the global sleep schedule. When a sink node receives this request, it will update the sleep schedule to increase the maximum level to the required value.

Each slot is divided into four portions (see Figure 5.11): clock drift compensation, SYNC period, REGISTER/DATA period, and the sleep period. The clock drift compensation is calculated using a maximum deviation of 30 ppm ($T_{MAX} = \pm 2\delta T$, where $\delta = 30\mu$ and T is the time offset until the scheduled slot). The SYNC period has a fixed base value of 20 ms and can dynamically increase using the conservative timeout activation technique. The REGISTER/DATA period also has a base value of 20 ms and an adjustable portion based on traffic patterns, also using the conservative timeout activation. After the REGISTER/DATA period, a node will go to sleep until the next scheduled slot.

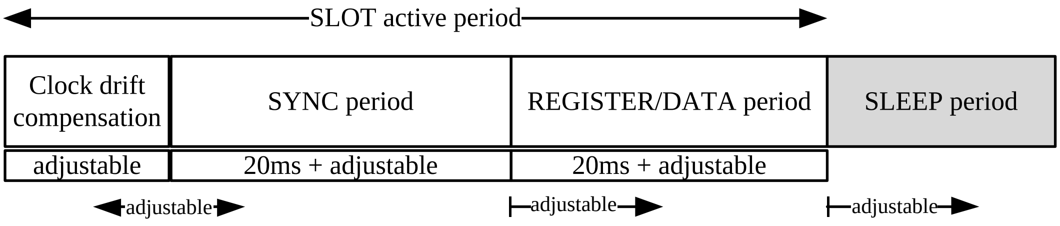


FIGURE 5.11: Slot.

5.4.3 Clear channel assessment

The clear channel assessment (CCA), shown in Figure 5.12, is an important part of CSMA protocols: it helps to reduce the number of collisions. The CCA process will start by a node performing a carrier sensing operation: if the channel is clear it will wait for the duration of a contention window (a uniformly distributed time of between 0 ms and 1 ms), if a carrier sensing operation finds the channel to be busy, it will reschedule the CCA 1 ms in advance and try again. A second carrier sensing operation is performed after the contention window, if the channel is still idle the CCA concludes that the channel is clear. This process will be repeated until the CCA results in a clear channel.

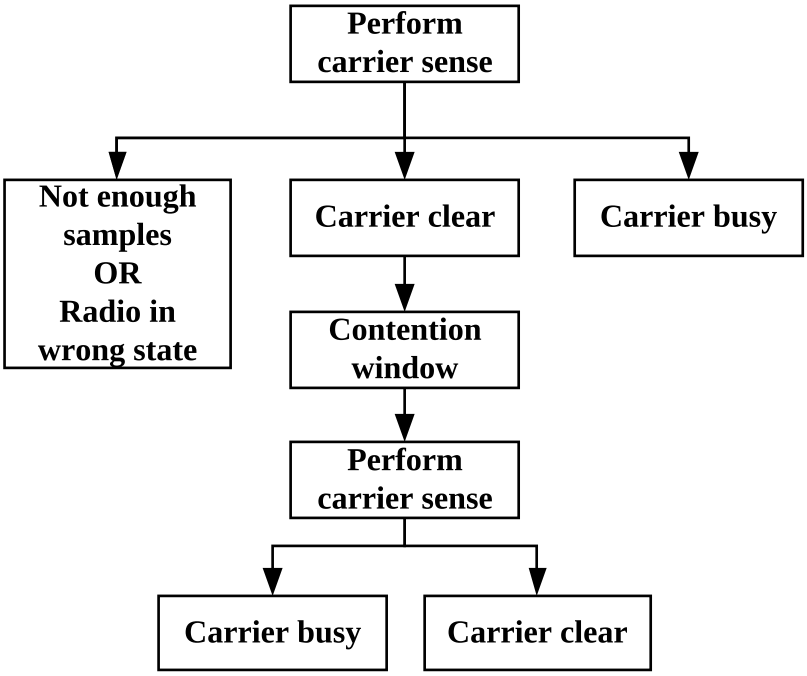


FIGURE 5.12: Clear channel assessment.

5.4.4 MAC packets formatting

All MAC packets require the following information: source, destination, type and frame check sequence (FCS). The source indicates who originated and sent the packet. The destination indicates which node should accept and process the packet. The type field indicates the type of the packet. The FCS is used for basic error detection.

The synchronisation beacon (SYNC) packet (Figure 5.13) is used to broadcast a sleep schedule to all nearby nodes. The destination address will always be set to the broadcast address. Each SYNC packet include the geographical coordinates (used for routing purposes) of the source node, and its own sleep schedule. The sleep schedule is constructed as follows: currentLevel (1 byte), maxLevel (1 byte), myParent (2 bytes), routingCost (2 bytes), and sleepOffset (4 bytes). The myParent variable identifies the parent node to whom the current node aggregates data to. The sleepOffset variable is used for offset time synchronisation. The maxLevel refers to the global sleep schedule network depth. The currentLevel is used to indicate if a REGISTER packet should be forwarded. The routingCost is the additive cost for routing to the sink. Each node synchronises with its parent node.

Source	Destination	Type	Coordinates	Sleep schedule	FCS
4 bytes	4 bytes	1 byte	4 bytes	10 byte	1 byte

FIGURE 5.13: SYNC packet.

The REGISTER packet (Figure 5.14) is used during network convergence. A REGISTER packet is generated when a node wishes to join the network but the required level does not exist. Whenever the maximum level of the current global sleep schedule is less than the required level, a node will flood a REGISTER packet, requesting to increase the maximum level of the schedule.

Source	Destination	Type	Required level	FCS
4 bytes	4 bytes	1 byte	1 bytes	1 byte

FIGURE 5.14: REGISTER packet.

The RTS packet (Figure 5.15) and the CTS packet (Figure 5.16) has exactly the same format, only the type packet field will indicate the difference between packets. The network allocation vector (NAV) indicates the estimated communication time for the transfer and is calculated by adding up the transmission times of the RTS, CTS, DATA, and ACK packets. Virtual carrier sensing allow nodes to sleep, reducing energy wastage due to overhearing. Possible transmission interference is reduced by deactivating nodes while other nodes are communicating.

The DATA packet (Figure 5.17) is used to transfer sensing information to the sink node. The size of the DATA packet is adjustable, each node aggregates data for an

Source	Destination	Type	nav	FCS
4 bytes	4 bytes	1 byte	4 bytes	1 byte

FIGURE 5.15: RTS packet.

Source	Destination	Type	nav	FCS
4 bytes	4 bytes	1 byte	4 bytes	1 byte

FIGURE 5.16: CTS packet.

unknown number of nodes. The sample size has been set to 16 bytes for simulation purposes, but the formatting of data samples will be application specific.

Source	Destination	Type	DATA sample(s)	FCS
4 bytes	4 bytes	1 byte	n x 16 bytes	1 byte

FIGURE 5.17: DATA packet.

After each DATA packet, a node will reply with an ACK packet (Figure 5.18). Link-layer acknowledgements ensure end-to-end reliability. If a node does not receive an ACK packet for transmitted DATA, it will restart the unicast transmission process.

Source	Destination	Type	Sequence number	FCS
4 bytes	4 bytes	1 byte	4 bytes	1 byte

FIGURE 5.18: ACK packet.

5.5 NETWORK SPECIFICATION

5.5.1 Geographical obtuse triangulation routing

The novel geographical obtuse triangulation routing protocol is based on a Bellman-Ford distance-vector, an additive cost function algorithm which scales well for large networks. Only local neighbourhood information is necessary in distance vector algorithms, link-state algorithms require the entire network topology to be known which creates scalability problems in WSNs.

Figure 5.19 illustrates the routing protocol process where node A has to make a routing decision: must it use node B or node C to reach the data sink? The geographical obtuse triangulation routing protocol operates as follows:

If $\Theta > 90°$ (obtuse triangle) use node B, otherwise use node C. For an obtuse triangle the following holds true: $ac^2 > ab^2 + bc^2$. Thus, using $distance^2$ as the link cost metric validates the $\Theta > 90°$ criteria with extremely low complexity.

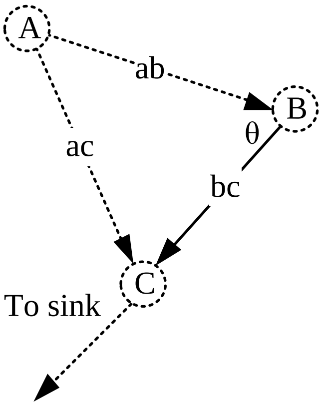


FIGURE 5.19: Geographical obtuse triangulation routing.

The basic operation of the presented routing protocol can be seen in Figure 5.19. Node C provides a routing path towards the data sink. Node B uses node C to get to the sink. Node A receives synchronisation beacons from both node B and node C and has to decide which to use as a parent node. Node A routing decision: $A \rightarrow B \rightarrow C$ OR $A \rightarrow C$. Criteria: $\Theta > 90°$. For $A \rightarrow B \rightarrow C$ the routing cost is: $C_{ADVERTISED\_COST} + bc^2 + ab^2$. For $A \rightarrow C$ the routing cost is: $C_{ADVERTISED\_COST} + ac^2$. When $\Theta > 90°$, then $ac^2 > ab^2 + bc^2$ which results in the $A \rightarrow B \rightarrow C$ data path.

5.5.2 Split horizon

Split horizon is a loop-prevention mechanism in distance-vector based routing protocols. The split horizon rule prohibits a node from advertising a route back in the direction from which it was learned. Split horizon is not normally applied in medium contention networks, a modification was introduced to adjust for wireless communication.

Synchronisation packets broadcast routing information using geographical coordinates to determine link cost and a sleep schedule which include the additive routing cost and sleep schedule information (including the maximum level, current level, and parent node). The modified split horizon rule is implemented as follows: a node must ensure that it filters out any SYNC packets where it is indicated as the parent node. This feature ensures that no routing loops are formed.

CHAPTER 6

SIMULATION EXPERIMENTS

Objectives of this Chapter

This section provides and explains the experimental procedures followed to simulate and validate the specifications of the proposed protocol. After this section, the reader should be familiar with the proposed experiments and the purpose of each experiment.

6.1 SIMULATION

The purpose of the simulation experiments is to act as a proof-of-concept for the designed protocol stack. Using the simulations, it can be determined whether the protocol design and implementation adheres to the design criteria and requirements.

6.1.1 Simulation platform

Castalia v3.2 [34] based on the OMNeT++ v4.2 [30]. For more information on Castalia, see Section 2.5.

6.1.2 Simulation parameters

Table 6.1 describes the default parameters used in simulations. Any deviation from a default setting will be included in the experiment description.

TABLE 6.1: Default simulation parameters of interest.

OSI layer	Parameter	Value
Radio layer (PHY)	Radio model	CC2420 - IEEE 802.15.4
	TX Output Power	0 dBm
	Carrier frequency	2.4 GHz
	Collision Model	Additive interference
Data link layer (MAC)	Adjustable SYNC period (minimum)	20 ms
	Adjustable DATA period (minimum)	20 ms
	Contention period	1 ms
	Listen timeout	10 ms
	Wait timeout	1 ms
	Slot time	10 s
	Maximum retransmissions	10
Application layer	Sampling period	300 s
	DATA packet size	16 bytes
	Deployment	Randomised (uniform)
	Deployment area	10 m x 10 m

6.2 EXPERIMENTS

In all experiments the black box approach, as seen in Figure 6.1, is used to inspect the effects of different input variables on certain output variables. The default simulation parameters have been set out in Table 6.1.

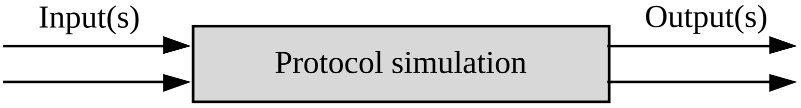


FIGURE 6.1: The black box analysis scheme.

In this section each experiment is described individually. Every experiment has a specific title, input variable(s) and output variable(s). A description is presented indicating why the experiment is needed and what it intends to validate. A prediction for each experiment proposes the expected findings.

6.2.1 Experiment 1

Title Simulation-time validation

Input variables Simulation time, number of nodes

Output variables Sleep schedule duty cycle (SSDC), SSDC variability and SSDC confidence intervals

Description The purpose of this experiment is to determine how long a simulation should run to provide a stable average sleep schedule duty cycle output value. Simulating a network for too short will provide inaccurate results, while simulating a network for too long does not have any negative effects but requires more processing time.

Prediction The average duty cycle should start off at an average of 1 % and then gradually lower until an equilibrium point is reached. The 1 % duty cycle is consistent with the fixed duty cycle during the setup phase. The equilibrium point should be unique to the size of the network.

6.2.2 Experiment 2

Title Network size effect on network convergence

Input variables Number of nodes

Output variables Network convergence time

Description The purpose of this experiment is to determine the expected convergence time for different size networks. Because of the random deployment scenarios and parameters, the convergence time cannot be modeled analytically.

Prediction The network convergence time is expected to increase with the number of nodes present. Variability in the results are expected: the deployment of nodes are randomised. The network convergence represents the time it takes for all nodes to synchronise and join the global sleep schedule.

6.2.3 Experiment 3

Title Network size effect on sleep schedule duty cycle

Input variables Number of nodes

Output variables Sleep schedule duty cycle

Description The purpose of this experiment is to validate the sleep schedule duty cycle design specification.

Prediction A sleep schedule duty cycle of below 1 % should be achieved, as per protocol design. The average sleep schedule duty cycle is expected to increase with larger network sizes because of the additional overhead, data and interference.

6.2.4 Experiment 4

Title Frame size and link reliability effect on latency

Input variables Frame size (number of slots), link reliability

Output variables Data reporting latency

Description The purpose of this experiment is to find the maximum latency boundary for a given frame size and the link reliability. The experiment should provide worst-case scenarios where the protocol will still adhere to the reporting latency requirement.

Prediction The allowable network depth will increase as the link reliability increases. An exponential curve is expected and the maximum depth cannot exceed 30.

6.2.5 Experiment 5

Title Network size effect on sent packets

Input variables Number of nodes

Output variables Sent packets breakdown

Description The purpose of this experiment is to evaluate the network protocol overhead of the synchronisation beacons (SYNC packets) by comparing the bandwidth utilisation in relation to DATA packets, the medium reservation (RTS/CTS packets), and the link layer reliability (ACK packets) overhead.

Prediction The expected result is that the medium reservation and reliability overhead will increase with the number of nodes. The expected increase would correlate with the increased data flow and interference and in larger networks. A data packet generated by a node situated in the last time slot will have to transverse all slots to reach the data sink.

6.2.6 Experiment 6

Title Network size effect on received packet interference

Input variables Number of nodes

Output variables Received packet interference

Description The purpose of this experiment is to evaluate the effectiveness of the CCA implementation with the added virtual carrier sensing feature. The CCA and virtual carrier sensing aims to reduce collisions.

Prediction The expectation is that interference will increase with a larger network. The goal is to produce a physical layer reliability of above 90% (excluding the contribution of the implemented link layer reliability mechanisms).

6.2.7 Experiment 7

Title Deployment area size effect on the sleep schedule duty cycle

Input variables Deployment area size

Output variables Sleep schedule duty cycle

Description The purpose of this experiment is to determine whether the deployment area affects the sleep schedule duty cycle. With larger deployment areas the nodal density decreases (when keeping the number of nodes constant) which reduces interference but increases the likelihood of hidden terminal problems.

Prediction Due to the geographical nature of the proposed protocol and medium reservation mechanisms the expectation is that a small effect in the average sleep schedule duty cycle will be contributed to varying the deployment area size.

6.2.8 Experiment 8

Title Network size effect on reliability

Input variables Number of nodes

Output variables Link layer reliability

Description The purpose of this experiment is to calculate the expected reliability based on the network size. The expected reliability is useful when generating the maximum latency model that determines the maximum frame length. Five different reliability measurements are extracted: *sent DATA packets vs received ACK packets*, *received CTS packets vs received ACK packets*, *sent CTS packets vs received DATA packets*, *sent RTS packets vs received ACK packets* (overall reliability), and *sent RTS packets vs received CTS packets.*

Prediction Increasing the number of nodes should decrease the link layer reliability, more nodes are contesting for the same wireless medium. The aim is to provide a link layer reliability of above 90 %.

CHAPTER 7

RESULTS AND DISCUSSION

Objectives of this Chapter

This chapter exhibits the results obtained from the experimental simulations which were described in Chapter 6. Every experiment is discussed in detail and the results are contextually evaluated. At the end of this chapter a discussion is presented which verifies the design optimisation goals and protocol specifications.

7.1 SIMULATION RESULTS

7.1.1 Experiment 1 - Simulation-time validation

A confidence interval of 99 % and a variability interval of 99 % were used as criteria for this simulation. A 99 % confidence interval (CI) signifies that the true average of the parameter under investigation lies within the span $[(sampleAverage - CI) \ldots (sampleAverage + CI)]$ for 99 % of the possible sample sets chosen. The variability interval (VI) is the tightest interval that includes 99 % of the samples. Variability intervals give an indication of where most of the values are and how they relate to the average.

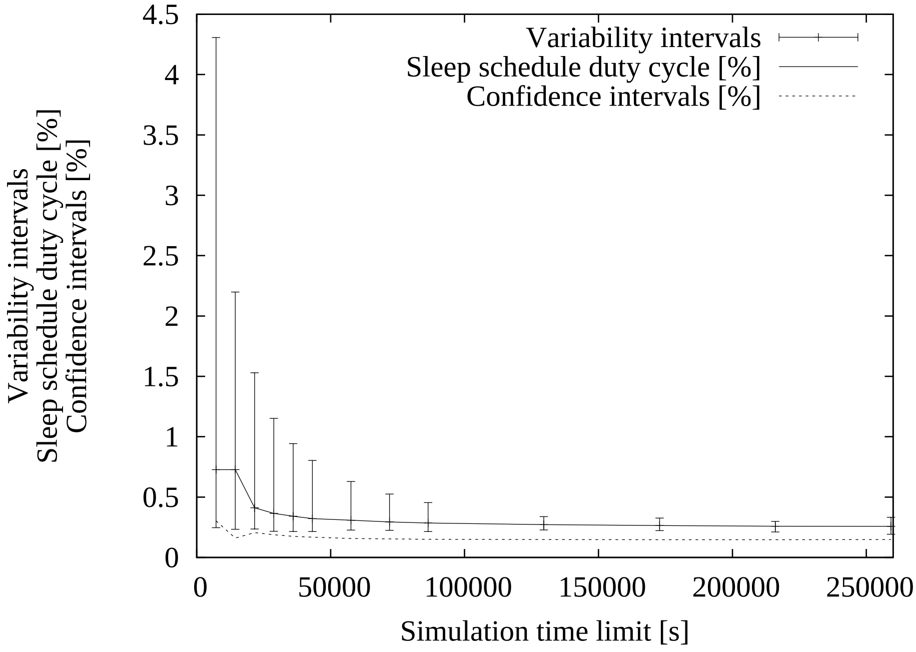


Figure 7.1: Results - Experiment 1 - Simulation-time validation. 10 nodes.

Figure 7.1 represents the simulation with only 10 nodes. From the results it is clear that there is little overall variability (below 1 % after 40,000 s simulation time) with the low number of nodes present, and the variability intervals decrease with longer simulation times. From this simulation the average duty cycle is constant after 50,000 s. However, the variability intervals show that the duty cycle finds an equilibrium point after 100,000 s simulation time. The confidence intervals are below 1 % from the start, which means that average duty cycle does not differ much between nodes.

Figure 7.2 shows the simulation with 100 nodes. In this simulation the increased variability by having more nodes in the network is illustrated, as predicted. It should be noted that the variability intervals are still extremely low (below 1 %). The confidence intervals indicate that the average duty cycle does not differ much (below 0.1 %) between nodes - this can be contributed to the larger number of nodes present and a very low variance. The duty cycle finds an equilibrium point after 100,000 s.

It was decided to run all simulations for 259,200 s (3 days) simulation time. From the results presented, it proved that the simulations will reach a steady state and will provide useful and relevant data.

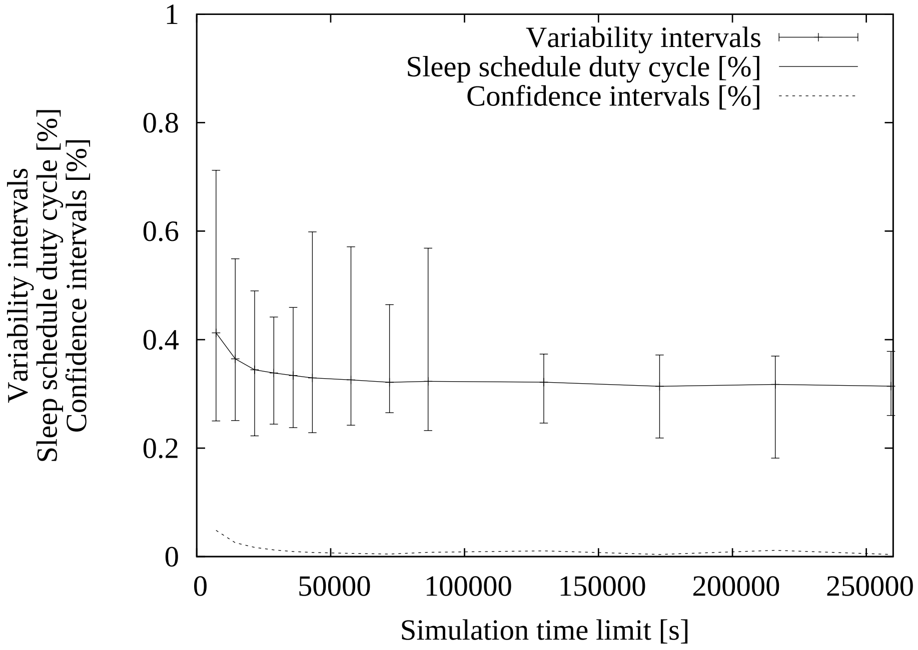


Figure 7.2: Results - Experiment 1 - Simulation-time validation. 100 nodes.

7.1.2 Experiment 2 - Network size effect on network convergence

Network convergence refers to the stable state reached when all nodes within the network has associated with the global sleep schedule. After network convergence all nodes have been assigned to a time slot and can perform data aggregation. Network convergence is an indication of how long it will take a network to be fully functional after it is deployed in the field.

From Figure 7.3 the output is seemingly linear. Each of the data points used in the graph is calculated as an average over 50 unique simulation repetitions. The network convergence is within acceptable limits. In the worst-case scenario (90 nodes) network convergence took just over an hour. This value can be reduced by increasing the duty cycle during the setup phase. But, the setup phase duty cycle was set to 1 % to adhere to the design considerations. When nodes cannot detect a data sink in a deployment (due to device failure), nodes should continue to use sleep scheduling (in the setup phase) and wait for another data sink to be deployed.

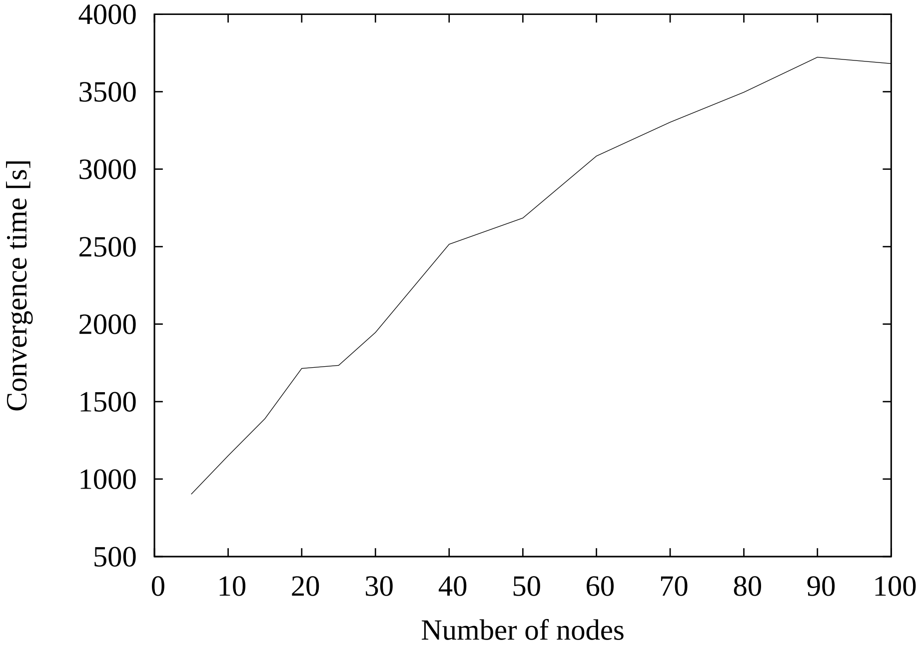


FIGURE 7.3: Results - Experiment 2 - Network size effect on network convergence.

7.1.3 Experiment 3 - Network size effect on sleep schedule duty cycle

The most important criteria for the design is the ultra-low sleep schedule duty cycle, and that the protocol should scale to networks of any size. In this experiment the sleep schedule duty cycle parameter is evaluated. The duty cycle is expected to be below 1 % for any number of nodes (up to a maximum limit of 100 nodes).

From the results presented in Figure 7.4 the average sleep schedule duty cycle is well below the 1 % design consideration. The results show that the duty cycle is close to 0.35 % and virtually no variability when increasing the size of the network. From these simulation results it can be concluded that the design, as implemented, demonstrates both energy efficiency and scalability and satisfies the optimisation goals.

7.1.4 Experiment 4 - Frame size and link reliability effect on latency

This experiment evaluates two of the research design objectives: the quality of service optimisation goal, and the latency protocol specification. The optimisation goal states that the design should ensure that traffic arrives at the sink in a timely manner. The

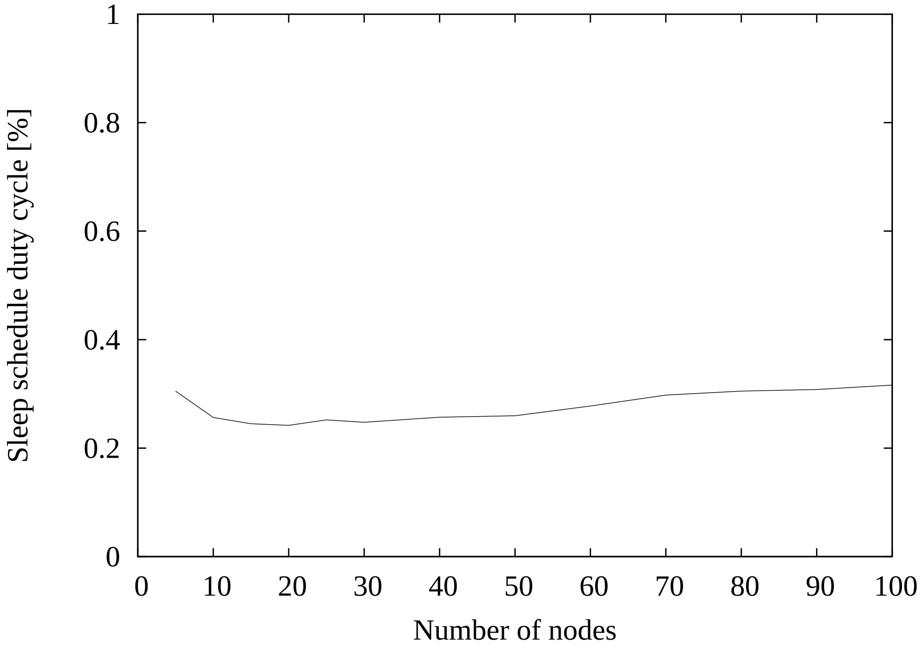


FIGURE 7.4: Results - Experiment 4 - Network size effect on sleep schedule duty cycle.

latency protocol specification requires that data should arrive at the sink before the originating node generates another measurement. In all experiments the data reporting period was defined as 5 min (300 s) and the slot time as 10 s.

The frame size and link reliability are not parameters that can be set, but are based on many different random variables. Thus, in this experiment a theoretical model is generated to simulate the maximum latency threshold:

$$\begin{aligned} Latency_{MAX} \approx\ & slotTime \cdot (maxLevel - 1) + \\ & slotTime \cdot maxLevel \cdot (1 - reliability^{maxLevel}) + \\ & slotTime \cdot 2 \cdot maxLevel \cdot (1 - reliability^{maxLevel})^2 + \\ & slotTime \cdot 3 \cdot maxLevel \cdot (1 - reliability^{maxLevel})^3 \end{aligned}$$

The first line of the equation refers to the inherent maximum latency of the slotted architecture with a link reliability of 1: when a node is allocated in the last slot and wants to transmit data, it transmits during the active portion of its parent node and the data is then aggregated in a staggered approach until it reaches the sink node (each slot takes *slotTime* to complete). Thereafter, each line represents the probability of a single/double/triple transmission failure. For a single transmission failure, the data reporting is delayed by an entire frame ($slotTime \cdot maxLevel$); a double transmission failure will delay the data by twice the frame length.

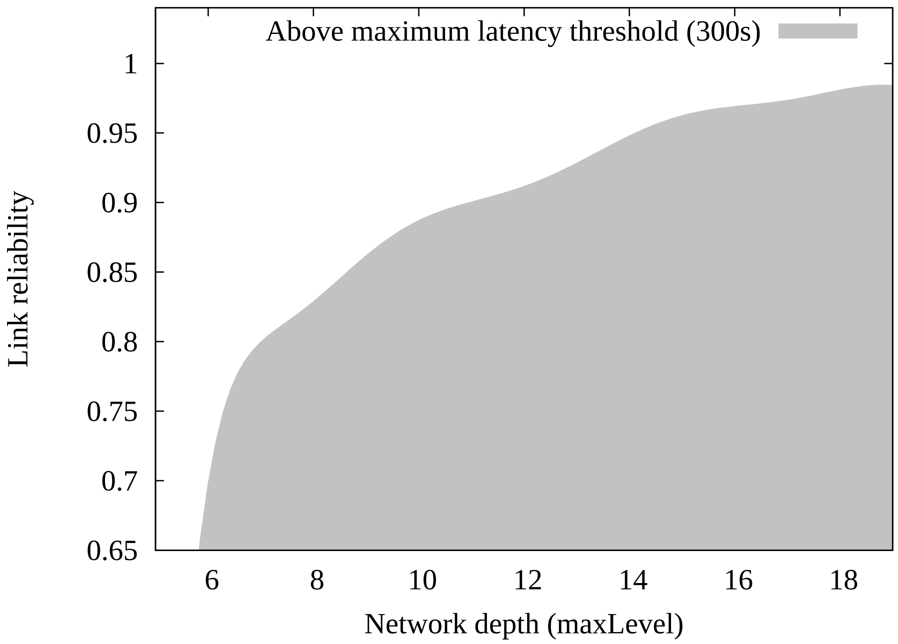


FIGURE 7.5: Results - Experiment 4 - Frame size and link reliability effect on latency.

In Figure 7.5 the latency threshold where the latency is above 300 s (the data reporting period) was generated. The latency threshold can be used during the design phase to limit the network depth when the link reliability is expected to be very low. Example: if it is known that the link reliability is 0.9 (90 %), the design should ensure that the network depth is 10 or less to satisfy the latency protocol specification requirement.

7.1.5 Experiment 5 - Network size effect on sent packets

This experiment compares the breakdown of sent packets based on the network size. The evaluation of the network overhead and bandwidth utilisation caused by periodic beacon messages are to be inspected. Figure 7.6 shows the average number of packets transmitted per node for the duration of 3 days.

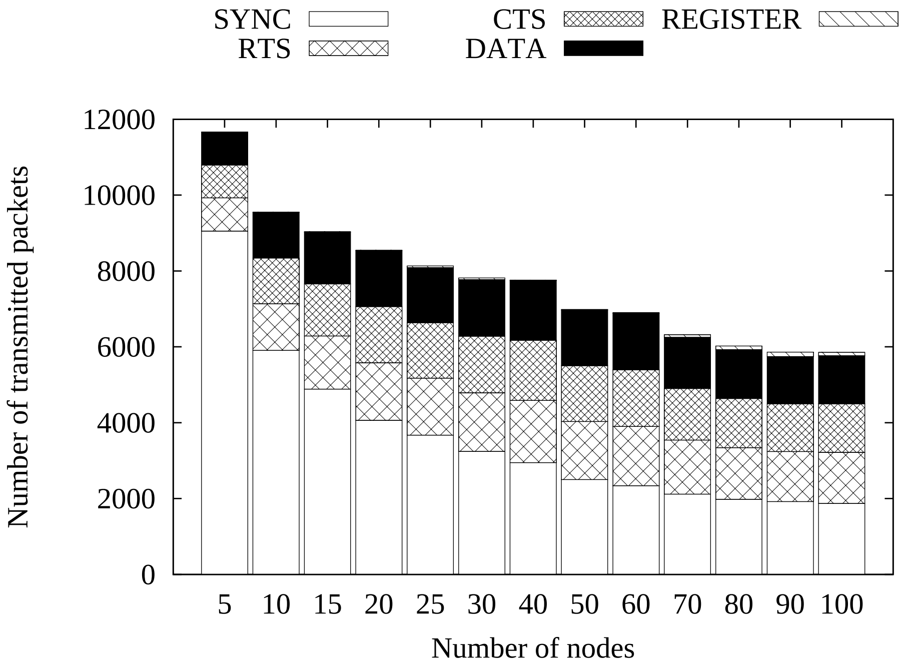


FIGURE 7.6: Results - Experiment 5 - Network size effect on sent packets.

From Figure 7.6 the average number of periodic beacon (SYNC) messages per node decreased as the number of nodes increased, this can be contributed to a larger network depth (more time slots resulted in each node having an active slot to transmit a beacon message less frequently). As the network size increases, the network overhead from beacon messages is reduced and the overall number of packets transmitted per node is decreased - this is favorable behavior, bandwidth utilisation is especially required in networks of larger sizes. The register packet is only transmitted during convergence and hardly makes an impact on the sent packet breakdown.

These results do not represent the true transmission bandwidth utilisation as data aggregation and the actual packet sizes are not taken into consideration. To calculate the true transmission bandwidth utilisation the size of packets need to be taken into consideration and not only the number of packets transmitted.

7.1.6 Experiment 6 - Network size effect on received packet interference

All nodes in the network utilise the same wireless spectrum which causes signal interference. Interference directly affects the reliability of the network. Interference occurs when multiple nodes try to simultaneously transmit: depending on the signal strengths of transmission, a collision can corrupt all data or in some cases data can still be retrievable. Different medium-access control methods are used to reduce collisions and can be classified into: fixed-assignment, demand-assignment, and contention-access protocol categories. The proposed design uses a slotted architecture (fixed-assignment) and CSMA/CA (contention-access) with virtual carrier sensing to reduce interference. This simulation examines the received packet interference and how it is effected with larger network sizes. The aim is to verify that the Clear Channel Assessment (CCA) works effectively to reduce the interference.

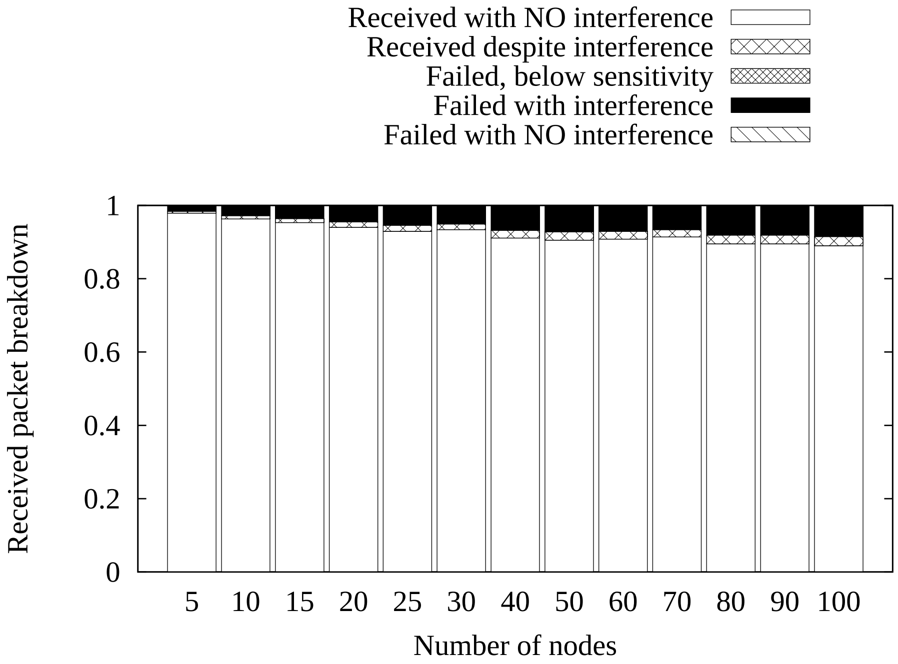


FIGURE 7.7: Results - Experiment 6 - Network size effect on received packet interference.

As expected the packet interference increased as the number of nodes increased, shown in Figure 7.7: when more nodes contend for the same wireless spectrum, collisions are more likely to occur. The results show that even for 100 nodes the successful reception rate (*received with no interference* and *received despite interference*) is above 90 %. These results concluded that the CCA is working well. When compared to a CCA implementation without the random contention window (independent tests during development which are not included in the dissertation) the successful reception rate was less than 60 %.

7.1.7 Experiment 7 - Deployment area size effect on sleep schedule duty cycle

The deployment area and number of nodes determine the nodal deployment density. The default deployment area for the simulations was a 10 m by 10 m (square) deployment region. The default area size was chosen in such a way that all nodes are deployed within communication range of each other: this simulates the worst-case scenario for interference where nodes are densely deployed. In this simulation, the square deployment region was increased while using a randomised deployment. To ensure connectivity for the increased deployment region, the number of nodes was set to 100 (without full connectivity some nodes will be stuck in the setup phase).

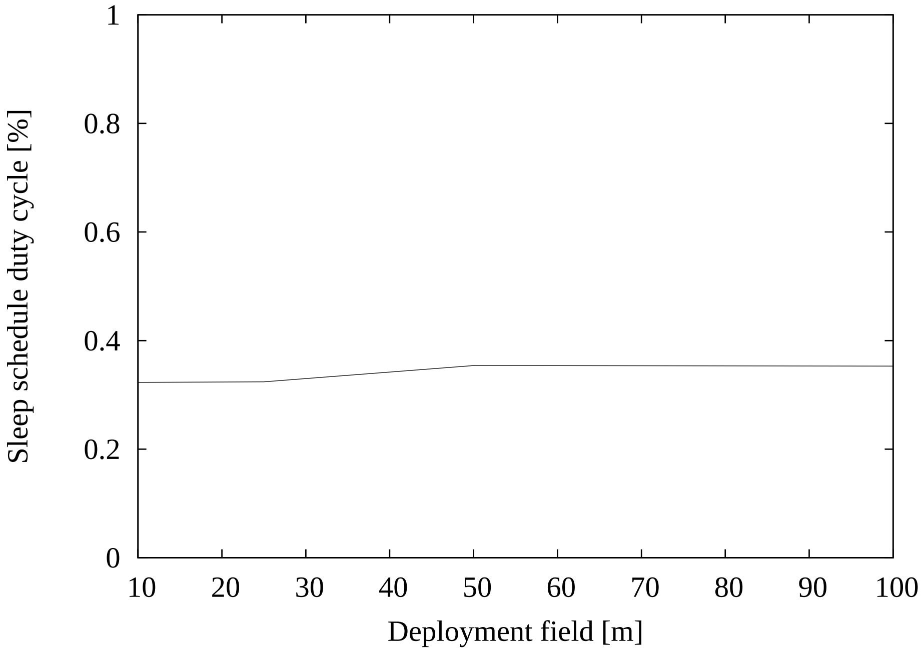


FIGURE 7.8: Results - Experiment 7 - Deployment area size effect on sleep schedule duty cycle.

As predicted, there is virtually no effect on the average duty cycle by changing the deployment area size (Figure 7.8). The geographical nature of the protocol and medium reservation mechanisms allowed for larger deployment areas without affecting the energy efficiency.

7.1.8 Experiment 8 - Network size effect on reliability

This experiment examines the network size effect on reliability. Various measurements are used to test the reliability of the network: *sent RTS vs received CTS*, *sent RTS vs received ACK*, *sent CTS vs received DATA*, *received CTS vs received ACK* and *sent DATA vs received ACK*.

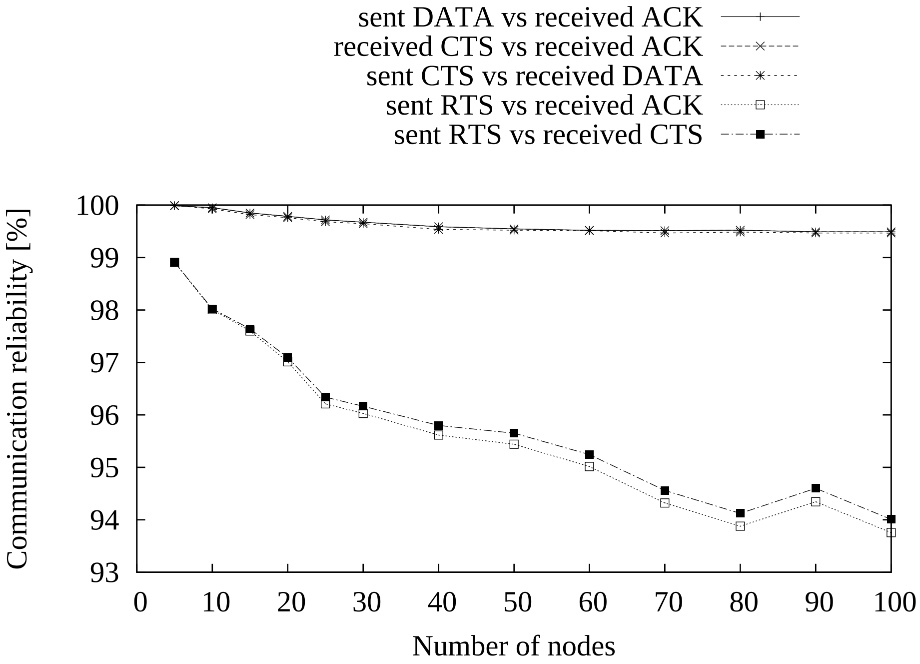


FIGURE 7.9: Results - Experiment 7 - Network size effect on reliability.

Figure 7.9 shows the results of the reliability simulation. The overall communication reliability is shown by *sent RTS vs received ACK* (this includes the whole RTS/CTS handshake, data transfer and link-layer acknowledgement). The overall communication reliability for 100 nodes is above 93 %, which can be considered as extremely good. Consideration should be taken that a node will retry 10 times within the same active period to resend a RTS packet; the retransmissions are not taken into account in the results presented here, each RTS packet represent a new communication attempt. These results show that the RTS has the largest probability to be lost due to interference, even though no CCA operation is performed for CTS, DATA, or ACK packets.

The overall reliability of 93.5 % can be increased to 99.5 % when a single retry attempt is allowed, and increased to 99.97 % when two retry attempts are allowed. Using 10 retry attempts the probability of successful transfers are basically 100 %.

7.2 DISCUSSION

The experiments in this chapter have proven that the proposed protocol adheres to the design objectives. The average sleep schedule duty cycle is well below the design objective of 1 %. The communication reliability is high (above 93 %) and the automatic repeat request (ARQ) feature ensures packet delivery. The high link reliability ensures that the protocol adheres to the QoS requirement. The protocol was proven to be scalable; the simulations have proven that the number of nodes present in the network does not adversely affect the performance of the protocol.

The objective of the research was to design an ultra-low duty cycle sleep scheduling protocol stack for wireless sensor networks. A detailed discussion on how the design requirements were achieved is presented below.

7.2.1 Optimisation goals

Quality of Service The results showed that the protocol provided significant QoS throughout the network. The CCA using a random contention window succeeded in providing reliable communication and the slotted architecture reduced interference by spreading transmissions over multiple time slots. A packet reception rate of above 93 % was achieved and the overall reliability is further improved using ARQ.

Energy efficiency A 1 % sleep schedule duty cycle can be achieved while still adhering to strict QoS requirements. Energy efficiency was improved by focussing on reducing the energy wastage caused by the major sources. Idle listening was reduced by using duty cycled sleep schedules and adaptive listening. Overhearing was reduced using virtual carrier sensing (NAV). Over-emitting was reduced by synchronising sleep schedules and compensating for clock drift. The network signalling traffic overhead was kept to a minimum during the design phase. Collisions were reduced by implementing a CCA using a random contention window.

Scalability The protocol was proven to be scalable. The number of nodes did not negatively impact the performance of the network, and the protocol stack performed well within the design protocol specifications for networks of up to 100 nodes. Currently the nodal population limit is unknown; given the current cost of hardware components, field implementations are not expected to be larger than 100 nodes.

Robustness Robustness is inherent in the design and was not simulated using an experiment. Routing entries are purged after 3 rounds if no synchronisation beacon is received. This feature is inherently tested during the network convergence phase.

7.2.2 Protocol specifications

Latency Data will arrive at the sink before the originating node generates another measurement if the design adheres to the link reliability and network depth threshold conditions.

Duty cycle A sleep schedule duty cycle of 0.35 % was achieved. This is well below the design specification of 1 %.

Self-configuration It was shown that the protocol allows network nodes to detect the presence of other nodes and to organise themselves into a structured, functioning network without human intervention.

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 CONCLUSION

The scope of the research was to develop an energy efficient and adaptive protocol stack for low data rate wireless connectivity with fixed devices and with very limited energy consumption requirements operating in a range of about 10 m. The research involved using various combined techniques to provide efficient sleep scheduling and increasing the network lifetime while monitoring the trade-off between increased network lifetime and latency, bandwidth and reliability.

A protocol stack was developed that succeeded in achieving an ultra-low (below 1 %) sleep scheduling duty cycle. The research concluded that it is possible to create a protocol stack that achieves a high level of energy efficiency while still adhering to strict Quality of Service (QoS) requirements.

The design uses the following features: a global sleep schedule, geographical data gathering tree, TDMA-slotted architecture, CSMA/CA, CCA with a randomised contention window, adaptive listening using a conservative timeout activation mechanism, virtual carrier sensing, clock drift compensation, and error control.

8.2 FUTURE WORK

8.2.1 Physical layer considerations

The IEEE 802.15.4 PHY specification was chosen as the PHY layer because of the wide range of transceivers already available that supports it. For research purposes ultra-wideband (UWB) communication can be investigated as a replacement for the direct-sequence spread spectrum (DSSS) used in IEEE 802.15.4 PHY.

Adaptive transceiver transmission power can be investigated to reduce the energy footprint. A node can save energy by reducing its transmissions power to such a manner that it can only communicate with its close proximity neighbours.

Techniques that support the use of multiple frequency channels can improve performance by reducing interference and increasing the available bandwidth.

8.2.2 Data link layer considerations

The current MAC design has many parameters that need to be optimised: the initial SYNC period, the initial REGISTER/DATA period, the random contention window period, the activation timeout and the slot time duration. In the current implementation, a CCA is not performed before sending a CTS: the reasoning behind the decision was that the node that sent the RTS already reserved the channel, but for non-dense deployments a hidden-terminal problem can arise and a CCA should be implemented before sending a CTS.

The sleep schedule duty cycle can be lowered even further if nodes do not awake during all time slots: nodes only need to awake for slots which are of interest to them. Various techniques can be investigated that could lower the sleep schedule duty cycle: clustering techniques, load balancing, more aggressive sleep scheduling, reducing network overhead introduced by periodic beaconing, spatial-correlation coverage-based aggregation, and energy aware metrics.

Even though the IEEE 802.15.4 provides ultra-low duty cycles, it lacks an important feature: time synchronisation of schedules. Research can be done to improve on IEEE 802.15.4 by implementing virtual clustering schemes to support multi-hop global sleep scheduling.

8.2.3 Network layer considerations

Geographical routing metrics do not always perform well: it does not consider the existence of poor communication links. Faulty hardware or obstructions can severely degrade link quality which could have devastating effects in large aggregated data gathering trees. Additional routing techniques need to be investigated.

The use of hierarchical clustering schemes should be investigated as possible methods to further increase the energy efficiency by reducing the sleep schedule duty cycle. Other routing metrics should be investigated with the aim to minimise the required energy to route data to the data sink.

Multi-sink support needs to be investigated and explored. A multi-sink environment will allow for larger network sizes and thus improve scalability and performance. Multi-path support can enable the implementation of traffic load balancing.

8.2.4 Application layer considerations

The application layer has two variables: the sampling period (300 s) and the sample size (16 bytes). These parameters directly affect the bandwidth utilisation. Increasing the sample size, or reducing the sampling period, will increase the bandwidth utilisation which could have negative effects. The bandwidth constraint issue has not yet been addressed with the current experiments and needs to be investigated.

REFERENCES

[1] I. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, no. 8, pp. 102–114, 2002.

[2] K. Holger and A. Willig, *Protocols and Architectures for Wireless Sensor Networks.* John Wiley & Sons, 2005.

[3] M. Stemm and R. H. Katz, "Measuring and reducing energy consumption of network interfaces in hand-held devices," *IEICE Transactions on Communications*, vol. E80-B, no. 8, pp. 1125–1131, 1997.

[4] E. Shih, P. Bahl, and M. J. Sinclair, "Wake on wireless: an event driven energy saving strategy for battery operated devices," in *Mobile Computing and Networking*, 2002, pp. 160–171.

[5] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," in *Communications, 2002. ICC 2002. IEEE International Conference on*, vol. 5, 2002, pp. 3418–3423.

[6] A. El-Hoiydi and J.-D. Decotignie, "Wisemac: an ultra low power mac protocol for the downlink of infrastructure wireless sensor networks," in *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*, vol. 1, 2004, pp. 244–251.

[7] E. Lin and J. Rabaey, "Power-efficient rendez-vous schemes for dense wireless sensor networks," in *Communications, 2004 IEEE International Conference on*, vol. 7, 2004, pp. 3769–3776.

[8] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2002, pp. 1567–1576.

[9] T. van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, ser. SenSys '03. ACM, 2003, pp. 171–180.

[10] Y. Tselishchev, A. Boulis, and L. Libman, "Experiences and lessons from implementing a wireless sensor network mac protocol in the castalia simulator," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, 2010, pp. 1–6.

[11] G. Lu, B. Krishnamachari, and C. Raghavendra, "An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks," in *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, 2004, p. 224.

[12] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ser. WSNA '02. ACM, 2002, pp. 88–97.

[13] P. Bonnet, J. Gehrke, and P. Seshadri, "Querying the physical world," *Personal Communications, IEEE*, vol. 7, no. 5, pp. 10–15, 2000.

[14] J. Burrell, T. Brooke, and R. Beckwith, "Vineyard computing: sensor networks in agricultural production," *Pervasive Computing, IEEE*, vol. 3, no. 1, pp. 38–45, 2004.

[15] "Pods," 2011, http://www.botany.hawaii.edu/pods/. Last accessed in Feb 2011.

[16] "Corie," 2011, http://www.stccmop.org/CORIE/. Last accessed in Feb 2011.

[17] Z. Zhang, Q. Chen, T. Bergarp, P. Norman, M. Wikstrom, X. Yan, and L.-R. Zheng, "Wireless sensor networks for logistics and retail," in *Networked Sensing Systems (INSS), 2009 Sixth International Conference on*, 2009, pp. 1–4.

[18] S. Sultan, T. Khan, and S. Khatoon, "Implementation of hvac system through wireless sensor network," in *Communication Software and Networks, 2010. ICCSN '10. Second International Conference on*, 2010, pp. 52–56.

[19] L. Schwiebert, S. K. Gupta, and J. Weinmann, "Research challenges in wireless networks of biomedical sensors," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, ser. MobiCom '01. ACM, 2001, pp. 151–165.

[20] B. Latré, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," *Wireless Networks*, vol. 17, pp. 1–18, 2011.

[21] B. Krishnamachari, *Networking Wireless Sensors.* Cambridge University Press, 2005.

[22] "Tinyos," 2011, http://www.tinyos.net/. Last accessed in Mar 2011.

[23] "Contiki," 2011, http://www.sics.se/contiki/. Last accessed in Mar 2011.

[24] I. Akyildiz, T. Melodia, and K. Chowdhury, "A survey on wireless multimedia sensor networks," *Wireless Communications, IEEE*, vol. 51, pp. 921–960, 2007.

[25] M. Kuorilehto, M. Hännikäinen, and T. D. Hämäläinen, "A survey of application distribution in wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2005, pp. 774–788, 2005.

[26] *IEEE Std 802.15.4c-2009*, IEEE 802.15 WPAN Task Group 4, 2009.

[27] C. J. Leuschner, "The design of a simple energy efficient routing protocol to improve wireless sensor network lifetime," Master's thesis, University of Pretoria, 2005.

[28] "ns-2," 2011, http://nsnam.isi.edu/nsnam/index.php/Main_Page. Last accessed in July 2011.

[29] "ns-3," 2011, http://www.nsnam.org/. Last accessed in July 2011.

[30] "Omnet++," 2011, http://www.omnetpp.org/. Last accessed in July 2011.

[31] A. Sobeih, J. Hou, L.-C. Kung, N. Li, H. Zhang, W.-P. Chen, H.-Y. Tyan, and H. Lim, "J-sim: a simulation and emulation environment for wireless sensor networks," *Wireless Communications, IEEE*, vol. 13, no. 4, pp. 104–119, 2006.

[32] "Ssfnet," 2011, http://ssfnet.org/homePage.html. Last accessed in July 2011.

[33] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla, "Glomosim: A scalable network simulation environment," University of California, Los Angeles, Tech. Rep., 1999.

[34] "Castalia," 2011, http://castalia.npc.nicta.com.au/. Last accessed in Jun 2011.

[35] A. S. S. Park and M. B. Srivastava, "Sensorsim: a simulation framework for sensor networks," in *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, 2000, pp. 104–111.

[36] P. Levis, N. Lee, M. Welsh, and D. Culler, "Tossim: accurate and scalable simulation of entire tinyos applications," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, ser. SenSys '03. ACM, 2003, pp. 126–137.

[37] P. Levis and N. Lee, *Tossim: A simulator for tinyos networks*, 2003.

[38] "Mixim," 2011, http://mixim.sourceforge.net/index.html. Last accessed in July 2011.

[39] A. Kellner, K. Behrends, and D. Hogrefe, "Simulation environments for wireless sensor networks," Georg-August-Universität Göttingen, Tech. Rep., 2010.

[40] B. L. Titzer, D. K. Lee, and J. Palsberg, *Avrora: Scalable Sensor Networking Simulation with Precise Timing*, 2005.

[41] L. Girod, J. Elson, T. Stathopoulos, M. Lukac, and D. Estrin, "Emstar: a software environment for developing and deploying wireless sensor networks," in *In*

Proceedings of the 2004 USENIX Technical Conference, ser. ATEC '04. ACM, 2004, pp. 283–296.

[42] A. Prayati, C. Antonopoulos, T. Stoyanova, C. Koulamas, and G. Papadopoulos, "A modeling approach on the telosb wsn platform power consumption," *Journal of Systems and Software*, vol. 83, no. 8, pp. 1355–1363, 2010.

[43] *CC2420 2.4 GHz IEEE 802.15.4/ZigBee-ready RF Transceiver datasheet*, Chipcon, 2004.

[44] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2002, pp. 1567–1576.

[45] D. L. Mills, "Internet time synchronization: the network time protocol," *Communications, IEEE Transactions on*, vol. 39, pp. 1482–1493, 1991.

[46] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, ser. SenSys '03. ACM, 2003, pp. 138–149.

[47] M. Maróti, B. Kusy, G. Simon, and A. Lédeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, ser. SenSys '04. ACM, 2004, pp. 39–49.

[48] P. Spevak and P. Forstner, "Msp430 32-khz crystal oscillators," Texas Instruments, Tech. Rep., 2009.

[49] I. Akyildiz and M. C. Vuran, *Wireless Sensor Networks.* John Wiley & Sons, 2010.

[50] I. Stojmenovic, *Handbook of sensor networks: algorithms and architectures.* John Wiley & Sons, 2005.

[51] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "Macaw: a media access protocol for wireless lan's," in *Proceedings of the conference on Communications architectures, protocols and applications*, ser. SIGCOMM '94, 1994.

[52] K. Adere and G. Murthy, "Solving the hidden and exposed terminal problems using directional-antenna based mac protocol for wireless sensor networks," in *Wireless And Optical Communications Networks, 2010 Seventh International Conference On*, 2010, pp. 1–5.

[53] S. Singh and C. S. Raghavendra, "Pamas–power aware multi-access protocol with signalling for ad hoc networks," *ACM SIGCOMM Computer Communication Review*, vol. 28, pp. 5–26, 1998.

[54] C. Schurgers, V. Tsiatsis, and M. Srivastava, "Stem: Topology management for energy efficient sensor networks," in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 3, 2002, pp. 1099–1108.

[55] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 3, 2004, pp. 1534–1539.

[56] H. Pham and S. Jha, "An adaptive mobility-aware mac protocol for sensor networks (ms-mac)," in *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, 2004, pp. 558–560.

[57] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, "Protocols for self-organization of a wireless sensor network," *Personal Communications, IEEE*, vol. 7, no. 5, pp. 16–27, 2000.

[58] L. Bao and J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, ser. MobiCom '01, 2001, pp. 210–221.

[59] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," *Wireless Networks*, vol. 12, pp. 63–78, 2006.

[60] R. Ha, P. Ho, and X. S. Shen, "Cross-layer application-specific wireless sensor network design with single-channel csma mac over sense-sleep trees," *Computer Communications*, vol. 29, no. 17, pp. 3425–3444, 2006.

[61] A. Boukerche, X. Cheng, and J. Linus, "Energy-aware data-centric routing in microsensor networks," in *Proceedings of the 6th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems*, ser. MSWIM '03, 2003, pp. 42–49.

[62] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, pp. 325–349, 2005.

[63] J. Kulik, W. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," *Wireless Networks*, vol. 8, pp. 169–185, 2002.

[64] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *Networking, IEEE/ACM Transactions on*, vol. 11, no. 1, pp. 2–16, 2003.

[65] D. Braginsky and D. Estrin, "Rumor routing algorthim for sensor networks," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ser. WSNA '02. ACM, 2002, pp. 22–31.

[66] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, vol. 2, 2000, p. 10 pp.

[67] S. Lindsey and C. Raghavendra, "Pegasis: Power-efficient gathering in sensor information systems," in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 3, 2002, pp. 1125–1130.

[68] A. Manjeshwar and D. Agrawal, "Teen: a routing protocol for enhanced efficiency in wireless sensor networks," in *Parallel and Distributed Processing Symposium., Proceedings 15th International*, 2001, pp. 2009–2015.

[69] ——, "Apteen: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Parallel and Distributed Processing Symposium, Proceedings International, IPDPS 2002*, 2002, pp. 195–202.

[70] Y. Yu, R. Govindan, and D. Estrin, "Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks," University of California, Los Angeles, Tech. Rep., 2001.

[71] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, ser. MobiCom '01. ACM, 2001, pp. 70–84.

[72] V. Rodoplu and T. Meng, "Minimum energy mobile wireless networks," *Selected Areas in Communications, IEEE Journal on*, vol. 17, no. 8, pp. 1333–1344, 1999.

[73] L. Li and J. Halpern, "Minimum-energy mobile wireless networks revisited," in *Communications, 2001. ICC 2001. IEEE International Conference on*, vol. 1, 2001, pp. 278–283.

[74] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, "Speed: a stateless protocol for real-time communication in sensor networks," in *Distributed Computing Systems, 2003. Proceedings. 23rd International Conference on*, 2003, pp. 46–55.

[75] R. Fonseca, O. Gnawali, K. Jamieson, S. Kim, P. Levis, and A. Woo, "Tep 123: Collection tree protocol (ctp)," TinyOS Development, Tech. Rep., 2006.

[76] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09. ACM, 2009, pp. 1–14.

[77] M. Campista, P. Esposito, I. Moraesand, L. Costa, O. Duarte, D. Passos, C. de Albuquerque, D. Saade, and M. Rubinstein, "Routing metrics and protocols for wireless mesh networks," *Network, IEEE*, vol. 22, no. 1, pp. 6–12, 2008.