School of Computing and Communications
Lancaster University

Mitigating Interference Coexistence Issues in Wireless Sensor Networks

Alex King
B.Sc.

Submitted for the degree of
Doctor of Philosophy
October 2016

This project was supported by the Centre for Global Eco-Innovation and is part financed by the European Regional Development Fund

www.cgeinnovation.org

Abstract

Wireless Sensor Networks (WSNs) comprise a collection of portable, wireless, interconnected sensors deployed over an area to monitor and report a variable of interest; example applications include wildlife monitoring and home automation systems. In order to cater for long network lifetimes without the need for regular maintenance, energy efficiency is paramount, alongside link reliability. To minimise energy consumption, WSN MAC protocols employ Clear Channel Assessment (CCA), to transmit and receive packets. For transmitting, CCA is used beforehand to determine if the channel is clear. For receiving, CCA is used to decide if the radio should wake up to receive an incoming transmission, or be left in a power efficient sleep state. Current CCA implementations cannot determine the device type occupying the media, leaving nodes unable to differentiate between WSN traffic and arbitrary interference from other devices, such as WiFi. This affects link performance as packet loss increases, and energy efficiency as the radio is idly kept in receive mode.

To permit WSN deployments in these environments, it is necessary to be able to gauge the effect of interference. While tools exist to model and predict packet loss in these conditions, it is currently not possible to do the same for energy consumption. This would be beneficial, as parameters of the network could be tuned to meet lifetime and energy requirements. In this thesis, methods to predict energy consumption of WSN MAC protocols are presented. These are shown to accurately estimate the idle listening from environmental interference measurements.

Further, in order to mitigate the effects of interference, it would be beneficial for a CCA check to determine the device type occupying the media. For example, transmitters

may select back-off strategies depending on the observed channel occupier. Receivers could be made more efficient by ignoring all non-WSN traffic, staying awake only after detecting an incoming WSN transmission. P-DCCA is a novel method presented in this thesis to achieve this. Transmitters vary the output power of the radio while the packet is being sent. Receivers are able to identify signals with this characteristic power variation, enabling a P-DCCA check to reveal if the medium is currently occupied by WSN traffic or other interference. P-DCCA is implemented in a common WSN MAC protocol, and is shown to achieve high detection accuracy, and to improve energy efficiency and packet delivery in interference environments.

Declaration

This thesis is a presentation of my original research work. No part of this thesis has been submitted elsewhere for any other degree or qualification. All work is my own unless otherwise stated. The work was carried out under the guidance of Dr Utz Roedig, at Lancaster University's School of Computing and Communication.

28th October 2016

Alex King

Copyright ©2016 by Alex King.

"The copyright of this thesis rests with the author. No quotations should be published or information and results derived from this thesis without acknowledgement."

Acknowledgement

This thesis represents the culmination of four years as a PhD student at Lancaster University, School of Computing and Communications. It is the product of many good memories, of numerous highs and lows, and countless hours spent late into the night. Throughout, this experience has been shared with friends, colleagues, and family - to whom, this thesis is indebted.

First and foremost, this would not have been possible without the guidance, mentorship, and friendship of my supervisor, Professor Utz Roedig. From research skills to linux shell scripts, it has been a privilege to learn from Professor Roedig.

Secondly, for the constructive feedback, friendly advice, shared celebrations, and good friendships, I am grateful to the many colleagues I have worked with, in particular James, Ethem, Paul, and Martin; and to the many I have shared offices with, in particular Phil and Rajiv.

Thirdly, I am fortunate to have been funded - as part of a cohort of fifty PhD students - by the Centre for Global Eco-Innovation. Also, thank you to DemoPad Ltd, for sponsoring my PhD.

Fourthly, to my family. Without the love and support of my parents, Crawford King and Gillian Robson, I would not have had the confidence to pursue this endeavour - thank you for all that you've done, and everything that you do. I am also grateful to my Grandad, Anthony King - a programmer during most of his career, and one who has undoubtedly shared the frustrations and challenges reminiscent of a modern-day embedded systems programmer. Without your encouragement, I certainly would not have found a passion in computer programming, nor pursued a career in it.

Finally, and most importantly, thank you to my partner Deborah. You have shared with me all the highs and lows of this chapter of our lives, and I am jubilant as we draw it to a close together, and embark on the next.

Glossary

AP Access Point

BAN Body Area Network

BSS Basic Service Set

CAP Contention Access Period

CCA Clear Channel Assessment

CFP Contention Free Period

CPU Central Processing Unit

CRC Cyclic Redundancy Check

CS Carrier Sense

CSMA Carrier Sense Multiple Access

CTI Cross Technology Interference

DCCA Differentiating CCA

DCF Distributed Coordination Function

DIFS DCF IFS

DSSS Direct-Sequence Spread Spectrum

ED Energy Detection

FEC Forward Error Correction

FDMA Frequency Division Multiple Access

FP False Positive

GPIO General Purpose Input/Output

IBSS Independent BSS

IFS Inter-Frame Space

LPL Low Power Listening

LQI Link Quality Indication

LR-WPAN Low Rate Wireless Personal Area network

MAC Medium Access Protocol

MCU Microcontroller Unit

MD-DCCA Modulation Detection-DCCA

MWO Microwave Oven

NAV Network Allocation Vector

OFDM Orthogonal Frequency Division Multiplexing

OS Operating System

P-DCCA Power-DCCA

PAPR Peak to Average Power Ratio

PCB Printed Circuit Board

PRR Packet Reception Rate

RF Radio Frequency

RSS Received Signal Strength

RSSI Received Signal Strength Indicator

SDR Software Defined Radio

SFD Start of Frame Delimiter

SIFS Short IFS

SNR Signal to Noise Ratio

T-DCCA Time-DCCA

TDMA Time Division Multiple Access

TP True Positive

VCS Virtual Carrier Sense

WSN Wireless Sensor Network

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the 60 years following the development of the Integrated Circuit (IC), programmable computers have become widespread in many devices, from supercomputers to smartphones, and many applications, from military to healthcare. At the smaller, more discrete scale, advances in manufacturing techniques have allowed hardware to be built with a smaller footprint. Likewise, more efficient electronic design means devices can be built with lesser power requirements, permitting the use of more flexible power sources, including battery power, or energy harvesting. These trends have carved a path for embedded computer systems, which share many similarities with larger computers, but are typically smaller, more portable, and less powerful. Embedded systems are now found in mobile phones, MP3 players, engine control units, game controllers, and many other applications that require computation but have limited cost, space, or power requirements. In conjunction, wireless communication techniques have advanced in terms of range, data rate, and reliability. A number of physical and protocol standards are now available to meet numerous cost and functionality demands.

Wireless Sensor Networks (WSNs) are an example embedded system that has evolved over the past two decades in line with hardware advances and user demands. These consist of a network of sensor nodes, deployed in an environment to monitor a particular phenomenon. Each node is typically built around a Microcontroller Unit (MCU), incorporating the Central Processing Unit (CPU), memory, non-volatile storage, and

communication peripherals. The MCU interfaces with other on board devices, including wireless radio for node-to-node communication, power sources, and at least one sensor. Only limited data processing, such as compression or data aggregation, is done within the network. Instead, sensor readings are forwarded through the network toward a designated node, the sink, where they are made available for analysis. Example uses of WSNs are bridge structural health monitoring [KPC+07], wildlife monitoring [MCP+02], and volcano emission monitoring [WALR+06]. In all three cases, WSNs are preferred over more traditional sensors because of their smaller initial and operating costs, and in some cases because less human involvement reduces the risk of injury.

To allow flexible deployment, WSN nodes are often powered by a small, on board power source such as a battery. Likewise, WSN deployments may often be in environments where regular maintenance is difficult, making battery replacement to sustain a network either practically or commercially unfeasible. Therefore, the lifetime of the network is determined by the energy capacity and energy consumption of each sensor node. WSN nodes may also employ energy harvesting techniques such as solar panels. In this case, the energy consumption of each node must meet a limited budget.

On typical WSN hardware, the radio is the greatest source of energy consumption, even when not actively transmitting or receiving data. Over the past decade much research has focused on optimising the energy efficiency of WSN MAC protocols by keeping the radio powered down as much as possible. This conserves energy and extends the lifetime of the deployment. Since nodes can only communicate when their radios are powered-on, these MAC protocols must facilitate a synchronisation mechanism to send data. To meet energy efficiency requirements, while still able to transmit and receive, WSN nodes use Clear Channel Assessment (CCA) for both functions in the MAC protocol. CCA provides an indicator of the channel as busy or free, and is used by receiving nodes to detect incoming packets; and by transmitting nodes to avoid collisions.

In ideal conditions, these approaches allow nodes to reach extremely low radio duty cycles, enabling long network lifetime without sacrificing communication requirements. However, depending on the deployment environment, ideal conditions may not be possible

due to other interference sources. The $2.4Ghz$ Industrial, Scientific and Medical (ISM) frequency band is a common choice amongst WSN. This is shared with a plethora of other wireless networks and devices, including IEEE 802.11 (WiFi), 802.15.4 (ZigBee), and 802.15.2 (Bluetooth) standards. As well, other electronic devices - such as electric motors and microwave ovens - emit interference on this band as a by-product of their operation.

For most wireless networks, interference from other devices may cause packet decoding errors, which reduce link quality and throughput. This issue is referred to as Cross Technology Interference (CTI), and in WSN MAC protocols, is further exacerbated to also include a reduction in the energy-efficiency of the node. This stems from the use of CCA, which in most radio hardware is unable to discern WSN traffic from other interference. Consequently, when transmitting, WSN nodes are unable to adequately respond to channel contention from other interferers. This affects packet loss and the reliability of WSN links in these conditions. Under light interference, this may only negligibly impact WSN operation; under heavy interference, however, communication between nodes may be prevented entirely.

Similarly, under interference conditions the use of CCA during the receive process of WSN MAC protocols leads to false wakeups - where incoming traffic is wrongly inferred due to other channel activity. Energy is then wasted whilst the radio is kept powered on, increasing energy consumption beyond that expected in an ideal environment. Therefore, when installing a WSN in an environment subject to CTI, it is difficult to predict beforehand the energy consumption, and lifetime of the network.

Being able to estimate energy consumption, and also network lifetime, is an important tool for WSN designers: permitting more thorough feasibility assessments; and more efficient power supply design for the nodes. For example, given a lifetime requirement of at least six months, power supply components can be chosen more efficiently to meet this - knowing the estimated energy consumption. Alternatively, for the same hardware, WSN and MAC parameters can be tuned to meet minimum lifetime requirements.

Current state-of-the-art methods of energy estimation in literature are based on either

theoretical models, or network simulation. These estimations are able to account for numerous factors in WSN design, including hardware characteristics, node deployment, transmission range, routing and MAC protocol. For example, given the planned positions of each node and their hardware features, MAC and routing protocol parameters can be tuned appropriately to meet energy constraints. None of these methods are able to account for the effects of CTI, including false wakeups. Therefore, accurate prediction of energy consumption is not currently possible in environments subject to interference. This is a pivotal shortcoming in WSN deployment planning, due to the ubiquity of such environments. In this thesis, a methodology to estimate energy consumption is presented which accounts for false wakeups caused by CTI. This is based on MAC protocol behaviour and environmental interference measurements.

A number of solutions to mitigate the effects of interference on WSN MAC protocols have been proposed in literature. A taxonomy of these works is presented in this thesis, based on how the solution is implemented - divided into frequency avoidance, resilience, and detection approaches. These approaches offer numerous tradeoffs, in terms of hardware requirements, complexity, energy efficiency, and compatibility with other networks. Detection approaches - which bolster the channel sensing mechanism in order to detect other interference - are compatible with existing MAC protocols, and have been shown in previous literature to improve link quality and packet delivery performance. Current state-of-the-art solutions, however, incur the cost of significantly increased idle listening - reducing the energy efficiency even in interference free environments.

In this thesis, a detection-based solution is presented: Differentiating CCA (DCCA), which does not incur significantly increased idle listening, and is thus more energy efficient. DCCA is an extension to the standard IEEE 802.15.4 CCA, able to indicate the type of interference, as well as the current state of the channel. Power-DCCA (P-DCCA) is described as an implementation of DCCA, feasible on current WSN hardware. P-DCCA is evaluated in a common WSN MAC protocol, and shown to mitigate both energy inefficiency and packet loss in interference conditions.

Fuelled by falling manufacturing costs and advances in hardware design, commercial

and academic paradigms have spawned from WSN research to meet new demands. This includes Vehicular Sensor Networks (VSN), and Home Automation Networks (HAN), which share similar processing and communication requirements with WSN. The Internet of Things (IoT) proposes collection and exchange of data from ubiquitous embedded sources, across an existing infrastructure: typically, the internet. This is enabled by the advent of IPv6, whose large address space may allow each IoT device to be uniquely identifiable. In this thesis, the focus is specifically WSN, however the findings presented are applicable to these other domains also.

1.1 Problem Statement

Interference in the $2.4Ghz$ frequency domain can originate from co-located WiFi, Bluetooth, and microwave ovens, among other devices. In each case, coexistence with other technologies is often overlooked, resulting in sub-optimal performance for affected networks. This is the case for IEEE 802.15.4, whose link quality is degraded under CTI conditions. For WSN MAC protocols based on the IEEE 802.15.4 standard, this issue is further exacerbated in two regards:

- **Link Performance**

  Collisions between WSN and other interferers result in packet loss, which restricts the capacity of the link. This issue is made worse by the design of WSN MAC protocols, which are more susceptible to CCA-collisions than other IEEE 802.15.4 devices. Link capacity is further penalised due to infrequent sender/receiver synchronisation opportunities in these MAC protocols.

- **Energy efficiency**

  In interference environments, efficient use of the radio may cease due to two reasons. Firstly, in order to mitigate packet loss, retransmission and error correction mechanisms may be employed; these incur an additional energy cost. Secondly, interference is mistaken for incoming data in the MAC protocol wakeup sequence. This leads to the radio being left powered on, which wastes energy.

In this thesis, solutions are sought to two problems currently facing WSN deployments in interference environments, both of which stem from these issues raised above. These problems, P.1 and P.2 listed below, are referred back to throughout this thesis.

P.1 **Accurate energy consumption estimation of WSN**

Reliable prediction of WSN energy consumption, and therefore network lifetime, is a requirement not currently possible in environments subject to CTI, due to shortcomings in existing energy estimation methods. A mechanism to accurately predict node energy use and lifetime, which accounts for environmental interference, would be beneficial to WSN designers.

P.2 **Improved detection mechanism in WSN MAC protocols**

Current solutions to mitigating the effects of CTI that are based on WSN detection as part of the MAC protocol enable an improved interferer-collision response, improving performance under interference. Current solutions, however, require much greater idle listening time, reducing energy efficiency and battery life of each node. A detection-based solution which does not incur this penalty, while still able to mitigate interference effectively, would be beneficial to WSN deployments in interference environments.

1.2 Contributions

The contributions of this thesis, presented in chapters 4 and 5 are twofold, to meet the two issues raised above:

- Methods for estimating WSN MAC protocol duty cycle for a node, based on the environmental interference, are presented. This can be used to gauge node energy use and estimate network lifetime. Two methods are described and evaluated based on one typical WSN MAC protocol, ContikiMAC. This contribution is described as a tool available to WSN designers, to assist in pre-deployment design.
- An extension to the IEEE 802.15.4 standard Clear Channel Assessment (CCA) mechanism: Differentiating CCA (DCCA), which can differentiate between sources

of interference, as well as detecting the current occupation of the channel. Power-DCCA (P-DCCA) is an implementation approach to DCCA that is pursued in this thesis. This contribution is described as an approach to improve existing MAC protocol performance - in terms of energy efficiency and link quality - under interference.

1.3 Related Publications

Three peer reviewed publications have resulted from the presented work, in addition to another publication that is awaiting review. Each publication is listed below, including an explanation of how it relates to the work in this thesis.

- Alex King, James Brown and Utz Roedig. DCCA: Differentiating Clear Channel Assessment for Improved 802.11/802.15.4 Coexistence. In Proceedings of the 3rd International Workshop on Internet of Things Communications and Technologies (IoT - CT 2014). This workshop paper introduced the concept of DCCA in sensor networks, without any specific implementation.

- Alex King, James Brown, John Vidler and Utz Roedig: Estimating Node Lifetime in Interference Environments. In Proceedings of the 10th International Workshop on Practical Issues in Building Sensor Network Applications (IEEE SenseApp 2015). This workshop paper describes the work on estimating node lifetime in busy interference environments. This paper forms the basis of Chapter 4.

- Alex King, James Brown and Utz Roedig: Differentiating Clear Channel Assessment using Transmit Power Variation. This journal article has been submitted to the ACM Transactions on Sensor Networks (TOSN) journal. This paper forms the basis of Chapter 5. This journal article is awaiting review.

- Alex King, James Hadley and Utz Roedig: Dependability Competition: Contiki-MAC with Differentiating Clear Channel Assessment. This entry into the 13th International Conference on Embedded Wireless Systems and Networks dependabil-

ity competition incorporates P-DCCA into the ContikiMAC protocol. In this event, WSN MAC protocols were pitted against one another under interference conditions. P-DCCA came 6th out 11 overall, and scored the highest energy efficiency of all competitors.

1.4 Thesis Layout

The remainder of this thesis is divided into six chapters:

Chapter two presents the foundations of this thesis, beginning with typical WSN hardware and software components. This chapter then covers the IEEE 802.15.4 PHY and MAC wireless standard, which is a common choice in WSN design. Finally, sources of interference in the $2.4Ghz$ domain are described, and how they affect WSN.

Chapter three reviews the related work in the domain of IEEE 802.15.4 and WSN CTI. This includes firstly measurement studies and theoretical models, which evaluate the effects of interference on WSN. Then, previous solutions in literature to mitigating CTI are discussed and compared.

Chapter four presents a methodology for estimating energy consumption of a WSN in an interference environment, in order to meet the first problem, P.1, discussed in section 1.1. This includes a theoretical model and monte-carlo simulation of the WSN radio behaviour, in known interference conditions. By way of example, these techniques are demonstrated for a common WSN MAC protocol: ContikiMAC.

Chapter five focuses on the second problem, P.2, in section 1.1, and describes an extension to the standard IEEE 802.15.4 CCA mechanism: DCCA. Different implementation options of DCCA are compared, based on detection accuracy and energy cost. P-DCCA is presented as an implementation option that is available on commodity hardware, is capable of high accuracy, and is energy efficient. P-DCCA is implemented in a WSN MAC protocol, and evaluated in terms of link performance and energy efficiency. The evaluation includes small scale, single-hop deployments, and large testbed deployments spanning multiple hops.

Finally, the thesis is concluded in chapter seven. The contributions of this work

are reviewed in the context of the previous literature, and the scope for future work is considered.

Chapter 2

WSN Interference and Coexistence

In this section, the foundations of this thesis are presented. The history, evolving demands, and typical hardware of WSN are discussed in Section 2.1. WSN MAC protocols are discussed in Section 2.2. ContikiMAC, which is used as an example MAC protocol in Chapters 4 and 5, is described in Section 2.2.1. The effects of interference on WSN, and sources of interference, are discussed in sections 2.2.2 and 2.3 respectively. The chapter is summarised in Section 2.4.

2.1 Wireless Sensor Networks

The progression of manufacturing techniques, improved battery performance, and development of cost- and energy-efficient components catalysed the advance of WSN from academic to real world applications. WSN are well suited to applications requiring sensor mobility, such as inventory tracking and wildlife monitoring. In applications where regular maintenance is difficult, such as monitoring hazardous structures, or where pre-deployment planning of sensors is not possible, the self-organising and self-repairing nature of WSN is desirable. WSN are exemplified in scientific and industrial endeavours to monitor and report an environmental variable, such as wildlife tracking, volcano monitoring, oil well sensoring, and military deployments. More recently, the use case of WSN has expanded to include commercial applications, including Heating, Ventilation, and Air Conditioning (HVAC) systems, smart cities, and home automation. Notably, the

latter examples put the WSN on a collision course with other ubiquitous technologies sharing the frequency domain.

In most WSN applications, regular node maintenance - including battery replacement and recharging - is difficult or impossible. Therefore, the energy consumption of a node is an important design feature, dictating the lifetime of the network. Likewise, some node hardware designs use energy harvesting components, such as solar panels, to operate indefinitely. In which case, energy consumption is a limiting factor of the WSN, within which sensing, data processing, and communication must operate satisfactorily.

In WSN design, this requirement motivates processor selection in favour of energy efficient 8- and 16-bit microprocessors with memory typically in the tens of kilobytes, and Operating Systems (OSs) toward lightweight, event-based designs with little overhead. The work in this thesis adopts the Contiki WSN OS [DGV04], which provides a thread-based API familar to programmers [DSVA06], without the costly CPU overhead of pre-emptive scheduling. On-node processing is typically limited to data compression and aggregation, while more intensive processing is pushed beyond the sink node. The nature of the application dictates the choice of input sensors; examples include light, humidity, mechanical stress, acceleration, and movement sensors. For the typical WSN use case, which has little user interaction, output components are often limited to a few LEDs for status information.

The communication requirements in a WSN prioritise energy efficiency, the radio typically being the largest source of power consumption in node hardware. The traffic load in WSN is typically lower than other wireless applications; a simple deployment may need only to periodically report integer-value sensor readings, for example. The IEEE 802.15.4 specification is a suitable choice, more energy efficient than either 802.11 (WiFi) and 802.15.2 (Bluetooth), while still capable of an acceptable data rate and communication range for most WSN demands.

2.1.1 WSN Hardware

The work in this thesis was based around the Moteiv Tmote Sky [She04], which is now described. The processing, interface, GPIO and radio hardware of the Tmote Sky are comparable to many other WSN hardware examples. The Tmote Sky is based around an $8Mhz$ Texas Instruments MSP430 F1611 microcontroller [Ins09], with 10KB of RAM, and 48KB of flash storage. The board, measuring $3.2cm$ by $1.3cm$, includes an IEEE 802.15.4 $2.4Ghz$ physical layer-compliant TI CC2420 [Ins06] radio transceiver, with integrated Printed Circuit Board (PCB) antenna. The board also includes various integrated sensors and GPIO pins connecting to the MCU; programming is done over the USB connection by means of a pre-installed bootloader. The board also includes an FTDI UART-to-USB interface, which can be used for serial connection to a host computer.

The MCU communicates with the CC2420 radio via SPI. The CC2420 defines registers that configure and direct radio operation, such as the CCA thresholds, output power, and initiating a transmission. A single FIFO buffer is used for both sending and receiving packets, with a built-in Cyclic Redundancy Check (CRC) generator.

2.1.2 IEEE 802.15.4 (LR-WPAN)

The IEEE 802.15.4 Low Rate Wireless Personal Area network (LR-WPAN) is a common choice in WSN, and is therefore described in detail in this section. The standard defines wireless communication physical (PHY) and medium access control (MAC) layers for low power, low data-rate, and low cost applications making it suitable for WSN. The communication range of these devices is typically at least $10m$, described by the standard as the Personal Operating Space (POS). Suggested uses include Home Automation Systems (HAS), asset and inventory tracking, and industrial control. Unlike WLANs, little or no infrastructure is required for operation, which prioritises ease-of-installation and low maintenance overhead.

The standard defines multiple PHY options, each describing the operating frequency, and type of modulation. Each has tradeoffs between communication speed, range, security, and energy consumption. Four frequencies, and subsequent channels, are supported

based on the Industrial Scientific and Medical (ISM) bands: $780Mhz$, $868Mhz$, $915Mhz$, and $2.4Ghz$. Due to the number of channels, energy efficiency features, communication range, and data rate, the 2.4Ghz frequency is a common choice in devices using the LR-WPAN standard, including WSN. Here, two modulation methods are available: Chip Spread Spectrum (CSS) and Offset-Quadrature Phase-Shift Keying (O-QPSK), which provide $1Mb/s$ and $250kb/s$ date rates respectively. Most currently available $2.4Ghz$ IEEE 802.15.4 transceivers are based on O-QPSK, which has 16 channels, each $2Mhz$ wide, spaced $5Mhz$ apart. The work in this thesis was based on the 2.4Ghz O-QPSK PHY, although should be applicable to others as well.

The functionality provided by the radio is common across all PHY layers, and provides a common interface to the upper MAC and NET layers:

- **Received Signal Strength Indicator (RSSI)** an estimate of the received signal power on the channel, intended to be used by the network layer for the purposes of channel selection. No attempt is made to identify or decode signals on the channel. The RSSI result is calculated by averaging over eight symbol periods, which the $2.4Ghz$ O-QPSK defines as $128\mu s$. The standard requires that the span of RSSI values be at least $10dB$, and provide a linear relation between the power received in decibels to the RSSI value.

- **Link Quality Indication (LQI)** a characterisation of the quality of a received packet, which may be based on RSSI, Signal to Noise Ratio (SNR), or received symbol correctness. LQI is performed for each packet, and at least eight unique values should be supported. LQI is intended for the purposes of link quality assessment.

- **Clear Channel Assessment (CCA)** a determination of the current status of the channel as either *busy* or *clear*. This is intended for use in the Carrier Sense Multiple Access (CSMA)/CA mechanism, before transmitting data. The standard requires that at least one of the following CCA methods is supported:

  - *Mode 1: Energy Detection (ED)*

Figure 2.1: PHY Protocol Data Unit

If the energy detected on the channel exceeds the set threshold, which is configurable in software, a busy channel is reported.

– *Mode 2: Carrier Sense (CS)*

If a signal compliant with the standard, with the same modulation and spreading characteristics, is detected, a busy channel is reported. The signal may be above or below the ED threshold.

– *Mode 3: ED and CS*

A combination of both above methods, either of which may indicate a busy channel.

– *Mode 4: ALOHA*

The channel is always reported idle.

- **Channel frequency selection**: Transceivers must be able to change the current channel, within the allowable frequency for the particular PHY.
- **Data transmission and reception** Transceivers must be able to transmit and receive packets.

The 2.4Ghz PHY defines the PHY Protocol Data Unit (PPDU) as the underlying packet format, shown in figure 2.1, taken from [Ins06]. Each packet is prepended by a four-octet preamble, which is used for synchronising the receiver to incoming data. This is followed by the start of frame delimiter (SFD), a single octet with the value 0xE5, which indicates the start of a packet. The 7-bit frame length field describes the length of the PHY payload, and has a maximum value of 127, thereby dictating the maximum packet length for this PHY.

The IEEE 802.15.4 MAC protocol supports two network topologies: star, and peer-to-peer. The former has a single central node called a PAN coordinator, with which all other nodes exclusively communicate, and that is responsible for initiating and managing the network. This node has higher energy demands than other nodes, so is suited to applications that can afford a universal base-station, such as HAS and health care. Peer-to-peer topologies similarly have a PAN coordinator, but nodes are able to communicate amongst each other, therefore allowing multiple hops and more complex routing.

To cater for the diversity in supported devices, two device classes are defined in the standard: Full Function Device (FFD) - which can be used in either topology, and Reduced Function Device (RFD) - which can only be used in star topologies.

The MAC protocol supports two methods of channel arbitration: beacon-enabled, and nonbeacon-enabled. In beacon-enabled mode, the PAN coordinator broadcasts beacons which are used to synchronise attached devices and identify the PAN. The beacon is followed by the Contention Access Period (CAP), and Contention Free Period (CFP). Nodes wishing to communicate then use slotted-CSMA/CA within the CAP, else may be assigned Guaranteed Time Slots (GTS) within the CFP for uncontested channel access. This is useful in applications with specific bandwidth, or predictable latency requirements. Between beacons some quanta may be designated *inactive*, wherein nodes can sleep; this allows for some degree of energy efficient operation. In nonbeacon-enabled mode, nodes use unslotted CSMA/CA to mediate channel access between nodes. In both cases, acknowledgements are sent without CSMA/CA.

Nodes have two forms of address: a 64-bit long address, and a 16-bit short address which is assigned when the node joins the network. Each PAN is identified by a 16-bit PAN identifier, the selection of which is beyond the scope of the standard.

The standard defines the MAC Protocol Data Unit (MPDU), encapsulated within the PPDU, shown in figure 2.2, taken from [Ins06]. The 2-byte frame control field defines the packet type, addressing mode, and attributes required for processing the packet. The address fields may be either short (2-byte) or long (8-byte) as described above, and so the minimum length of the MPDU Header is 11 bytes. Each packet is appended with

<table>
<tr><td>Octets: 2</td><td>1</td><td>0/2</td><td>0/2/8</td><td>0/2</td><td>0/2/8</td><td>0/5/6/10/14</td><td>variable</td><td>2</td></tr>
<tr><td rowspan="2">Frame Control</td><td rowspan="2">Sequence Number</td><td>Destination PAN Identifier</td><td>Destination Address</td><td>Source PAN Identifier</td><td>Source Address</td><td rowspan="2">Auxiliary Security Header</td><td rowspan="2">Frame Payload</td><td rowspan="2">FCS</td></tr>
<tr><td colspan="4">Addressing fields</td></tr>
<tr><td colspan="7">MHR</td><td>MAC Payload</td><td>MFR</td></tr>
</table>

Figure 2.2: MAC Protocol Data Unit.

a Frame Control Field (FCS) to validate received packets. The FCS contains a 16-bit CRC, calculated over the MPDU contents.

There is a disparity between the goals of WSN applications and the IEEE 802.15.4 MAC protocol. While the latter strives for interoperability via standardisation, this is less prominent in WSN deployments, which favour flexibility, and have low likelihood of communicating beyond the WSN. WSN applications typically have requirements exceeding the IEEE 802.15.4 MAC, such as: low power consumption - which may demand more efficient radio usage, low latency - which may demand a more refined Time Division Multiple Access (TDMA) approach, and complex routing protocols - to support multi-hop networking. In light of this, most WSN designers opt instead for a WSN-specific MAC protocol, implemented in software, which is built on top of the IEEE 802.15.4 PHY interface. These are discussed in the next section.

The IEEE 802.15.4 physical layer is adopted by a number of other wireless standards, each of which define a separate MAC protocol. These include ZigBee [All09] and WirelessHART [wir], both of which are similarly affected by CTI. Consequently, the literature discussed in Chapter 3 is in most cases, unless stated otherwise, assumed interchangeable between these standards and WSN.

In the remainder of this thesis, the IEEE 802.15.4 standard is referred to as 802.15.4 for brevity. Unless stated otherwise, only the PHY layer is being referred to.

2.2 MAC Protocols

Given the implications of radio use on energy efficiency, many MAC protocols have been proposed for WSNs. As well, as WSN applications have diversified, so to have MAC

protocol requirements, to seek optimisation's also in QoS, link latency, and network throughput. An exhaustive review can be found in [DEA06, HXS+13]. In order to conserve energy, nodes duty cycle the radio between low power sleep and active modes, the latter required to send/receive data. Therefore, in order for two nodes to communicate, both must be in active mode.

In previous literature, the mechanism used to facilitate this synchronisation appears in two flavours: contention- and slotted/TDMA-based. Contention-based MAC protocols do not assume any synchronisation between neighbouring nodes' duty cycles. In order to send data, both nodes must first initiate a costly transmission mechanism. These approaches achieve a low idle cost, yet high transmission costs, and so are ideal in low data rate applications. Conversely, slotted approaches (also known as TDMA) require synchronisation amongst nodes in the network, each node assigned an uncontended slot to access the channel. This allows for more predictable latency, and high throughput under heavy load. When only a few nodes have data to send, the restriction on assigned transmission slots curtails maximum network throughput; this has been an active area of research in this domain.

Slotted approaches are preferable in high-data rate applications, however for infrequent communication, contention-based are more energy efficient, and therefore more common in typical WSN deployments. Contention-based approaches are therefore used as the basis for work in this thesis. The remainder of this section discusses the existing literature into contention-based MAC protocols.

Preamble sampling [EH02] is the precursor to many later contention-based MAC protocols which follow the same paradigm. Here, nodes periodically wakeup and check for channel activity, returning to sleep if an idle channel is detected. Transmissions are prepended with a preamble, of sufficient duration to be detected by the receiver. This approach allows nodes to keep the radio powered down most of the time, and achieve better energy efficiency under light traffic loads than previous TDMA-based protocols [SGAP00].

In either case, the idle listening cost is kept relatively low, at the expense of having a

high transmission cost: transmitters must transmit a preamble throughout the receivers' duty cycle, in order for a packet to be received. On the receiver-side, after detecting a preamble, nodes must wait upto the entire duration of the preamble before the packet begins. Also, other nodes besides the target node can detect this preamble. Energy is then wasted by keeping the radio powered until it is determined to be destined for another node; this is called the overhearing problem. El-Hoiydi et al describe WiseMAC [EHD04], a technique to reduce the transmitting cost. While running, each node in the WSN learns the wakeup schedules of its neighbours. Then, to transmit, the sender need only wait until the receiver is expected to wake up and sample the channel. Not only does this reduce the energy cost of transmitting, but also increases the throughput available to the channel. To reduce the overhearing problem, timing and address information can be encoded in the preamble, as in B-MAC+[PHC04]. Then, nodes can quickly determine the recipient of a packet, and non-target nodes can return to sleep, avoiding the overhearing problem. The timing information allows target nodes to discover the time remaining until the start of the packet, and power down until then.

Both approaches are incorporated into X-MAC [BYAH06], where the preamble is transmitted as multiple short packets strobes. Target nodes are able to acknowledge a strobe, thereby allowing the packet transmission to begin earlier, further reducing the cost of transmitting.

In many WSN MAC protocols, Low Power Listening (LPL) is adopted to reduce idle listening time and further improve energy efficiency. Here, CCA is used to infer incoming traffic [PHC04]. If the channel is detected busy, the radio is kept powered on to receive data; otherwise, the node enters sleep state. Thus, CCA are used in these protocols for both sending and receiving; this is illustrated in figure 2.3. LPL requires that the CCA threshold is set accordingly. Set too low, and false positives may be increased, causing the radio to waste time listening to an idle channel - referred to as a *false-wakeup*. Conversely, set too high and valid packets may be missed. In B-MAC [PHC04], this threshold is set based on prior measurements of the noise floor.

ContikiMAC [Dun11] is a LPL MAC protocol that is optimised to provide extremely

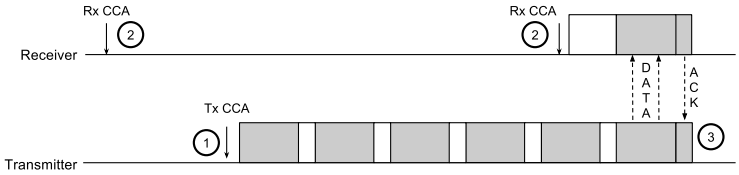


Figure 2.3: Operation of LPL MAC protocol

low duty cycles by optimising the idle listening mechanism. In this thesis, ContikiMAC is used as a base for optimisation in interference conditions, and is described in more detail in Section 2.2.1. Within the TinyOS Operating System, BoX-MAC [ML] is a similar LPL protocol referred to as LPL.

Receiver-initiated MAC protocols have also been proposed as an alternative approach to the transmitter/receiver handshake [SGJ08, DDHC+10]. Here, the receiver periodically broadcasts a beacon, then listens for incoming data. To send a packet, the sender listens to the channel for a beacon, then transmits the packet. This approach firstly reduces the channel use compared to preamble sampling, thereby leaving more channel bandwidth available to the rest of the network. Secondly, collisions can be handled more centrally by the receiver, which allows for a faster collision-resolution strategy. This approach does incur a higher idle listening cost, however has been shown to be more energy efficient under certain traffic loads than sender-initiated alternatives.

2.2.1 ContikiMAC

ContikiMAC [Dun11] is used in this thesis in Chapters 4 and 5 as an example LPL MAC protocol, to exemplify the solutions described therein. Therefore, the operation of ContikiMAC is described here.

Incorporating previous MAC protocol design techniques from literature, ContikiMAC is able to achieve a low duty cycle in ideal environments. Nodes periodically wakeup to sample the channel for activity. If detected, the radio is kept powered on to receive the packet, otherwise turned off to conserve energy. To send a packet, the sender repeatedly transmits the packet payload - referred to as packet strobes - throughout the receivers wakeup period. The senders' packet strobes must coincide with the receivers channel

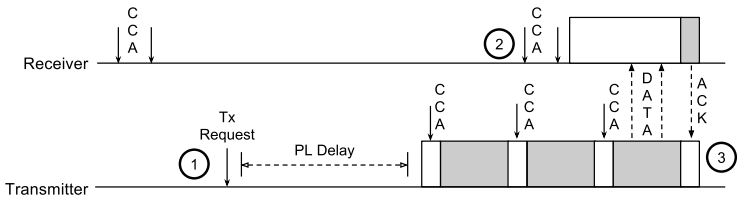


Figure 2.4: Normal Operation of ContikiMAC

check, stopping after a collision with another node, an ACK is received, or a timeout occurs.

CCA is employed in ContikiMAC for both sending and receiving. Senders sample the CCA six times before transmitting to ensure the channel is free, and sample once between each packet strobe, to detect an acknowledgement or a collision. To reduce the energy cost of transmitting, senders postpone the packet strobes until the receiver is expected to wakeup (similar to WiseMAC [EHD04]). This mechanism is called Phase Lock (PL).

To receive a packet, CCA is used for the periodic channel checks. In order to be confident that the CCA coincides with an ongoing strobe transmission, and not the period in between, two CCA are used and timed to ensure at least one can detect a packet strobe. This time window must be minimised to reduce the cost of idle listening. ContikiMAC therefore relies on hardware acknowledgements, optional in the 802.15.4 standard. By default, the time between transmissions is $400\mu s$, and the time between CCA checks is $500\mu s$. The operation of ContikiMAC is depicted in figure 2.4.

If the CCA erroneously detects other interference, [Dun11] describes *Fast Sleep*: if, after detecting incoming traffic, the channel remains quiet for a number of subsequent CCA, the receiver exits the wakeup sequence and powers down. Likewise, if no packet is received after a set number of CCA, which may also be due to a packet being destined to another node, the receiver returns to sleep.

2.2.2 Effect of interference

The $2.4Ghz$ ISM band is shared with a plethora of other wireless technologies. Among which, transmission powers, modulations, bandwidth, and channel arbitration policies

are far from homogeneous. Consequently in environments shared with other devices, link performance in these networks may be degraded. This issue is referred to as Cross Technology Interference (CTI), and is an important consideration in the design of any wireless protocol, and prior to any deployment.

Compared to other communication protocols, and standard 802.15.4 devices, the effects of CTI are more exaggerated in WSN MAC protocols, such as ContikiMAC. This is due to the infrequent nature of rendezvous opportunities, and the need to achieve low power operation. The effects of CTI are twofold on WSN: packet loss, and energy inefficiency.

Packet Loss

Packet loss is caused by collisions - simultaneous use of the channel - with other interference; this degrades network reliability and throughput. Packet loss may result from two types of interaction: packet-collision, and CCA-collision. Packet-collisions occur when another interference signal coincides with a WSN transmission. The reduced signal quality at the receiving node prevents the packet contents from being received correctly. This is the most common cause of packet loss in typical communication protocols.

CCA-collisions occur in communication protocols that implement a listen-before-send policy - including LR-WPAN, WSN MAC protocols, and IEEE 802.11. Here, nodes check the channel is free before transmitting using a CCA check. If the channel is deemed busy, due to an interference signal on the same frequency - the transmission is aborted. Depending on the retransmission policy, another attempt may be rescheduled.

The occurrence and effect of CCA-collisions are amplified in contention-based WSN MAC protocols. Unlike other 802.15.4 protocols, the transmission procedure in these protocols requires a lengthy synchronisation phase with multiple CCA checks - any one of which may cause a CCA collision. For example, in the ContikiMAC send sequence, six CCA checks are required before transmitting, followed by further CCA checks between each packet strobe. This is shown in figure 2.5. By contrast, the standard 802.15.4 MAC requires only a single CCA check in the CAP, and no CCA checks at all in the CFP. Therefore, as studied in previous literature ([BVT$^+$10] - discussed in Section 3.2), packet

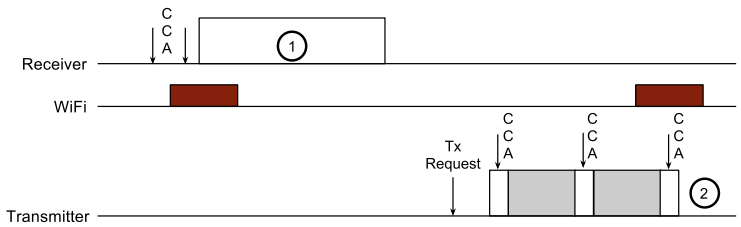


Figure 2.5: ContikiMAC under WiFi interference

loss due to interference is more prevalent in WSN-specific MAC protocols.

Energy inefficiency

Energy efficiency is an essential feature of WSN MAC protocols, in order to provide long network lifetime on minimal resources. Under interference conditions, energy consumption of WSN MAC protocols is increased. Therefore, network lifetime may be reduced compared to an idealistic environment.

Energy inefficiency may stem from two sources of energy use: transmitting or receiving, and idle listening. In the first instance, mechanisms to counter packet loss, such as retransmissions, error correction coding, and routing changes, consume more energy than in an interference-free environment. This may be necessary to achieve quality of service requirements in a deployment.

Secondly, the energy consumption of idle listening - when not receiving or transmitting data - is also influenced by interference. This is due to the listening mechanism during periodic wakeup checks, which use CCA to detect if another node may be transmitting. If this CCA detects other interference, a *false-wakeup* results, where the node enters the wakeup procedure, increasing energy consumption [ZCW+14]. For example, this is shown for ContikiMAC in figure 2.5. In typical WSN deployments, which have low data rate requirements, idle listening is the predominant source of energy consumption. Therefore, in interference environments, false wakeups are likely to be the greatest source of energy inefficiency.

2.3 Interference Sources

In this section, interference sources that are common in the $2.4Ghz$ frequency domain are discussed. This includes *communicable* devices, such as WiFi and Bluetooth, which define a standardised channel access protocol. Also, Microwave oven interference is discussed as an example of *non-communicable* interference source. In both cases, 802.15.4 networks are known to suffer in coexisting deployments.

2.3.1 IEEE 802.15.1 (Bluetooth)

Bluetooth is a PHY and MAC protocol for short range, high data rate communication originally intended to replace cables connecting electronic devices [Blu10]. Devices include wireless computer peripherals, such as mice and keyboards, wireless headphones and hands free headsets, and file transfer between smart phones, to name a few examples.

Bluetooth devices operate in the 2.4Ghz frequency domain, and use Frequency Division Multiple Access (FDMA) to mediate channel access. The channel is split up into 79 channels (in the US and most of Europe, 23 in Japan, Spain, and France), each $1Mhz$ wide. Nodes implement channel hopping - following a pseudo-random hopping sequence, known to all participating devices - to achieve robust communication in the presence of other interferers. Within each piconet, consisting of two or more devices, nodes are hop synchronised to communicate reliably. The hop duration is $625\mu s$, and nodes switch channels at a rate of 1600 hops/second. The basic specification supports a data rate of $721Kb/s$, while an enhanced data rate supports upto $2.1Mb/s$.

Unlike other 2.4Ghz protocols which implement a CSMA-style protocol, such as WiFi and ZigBee, Bluetooth does not have any CCA, or listen-before-talk paradigm. However, Adaptive Frequency Hopping (AFH) provides Bluetooth devices the ability to mark channels as *used* and *unused*, using only the former within the hopping sequence. Both master and slave devices within a piconet can detect interference, and ensure these channels are marked *unused*.

This coexistence approach is beneficial to other 2.4Ghz protocols, including 802.15.4. However, in a heavily interfered environment, Bluetooth is forced to use channels that

are affected by interference. Further, other networks with a low data rate or duty cycle - as is the case with WSN protocols - may not be detected by Bluetooth devices prior to establishing a piconet, and hence those channels will not be marked.

2.3.2 IEEE 802.11 (WiFi)

IEEE 802.11 was first released in 1997, to provide high speed LAN access to wireless devices. It is typically the last hop in a network topology, enabling network access to electronic devices requiring mobile network access, such as smartphones and laptops. The brand name for IEEE 802.11, including later amendments and revisions, is $WiFi$, which is used interchangeably in this thesis. Amendments to the protocol are provided as wireless technology progresses, increasing data rates and feature set. In the OSI model, these changes relate to the physical and data link layer, presenting the same interface to higher applications.

An 802.11 network consists of a number of addressable nodes, referred to in the standard as stations (STA). Each network consists of a Basic Service Set (BSS), as the basic building block, defining an area within which stations can communicate. An Independent BSS (IBSS) is the simplest form of BSS, consisting of two stations which can communicate directly. This is often known as an Ad-hoc network. By contrast, an Infrastructure BSS includes an Access Point (AP), which may provide access to a wider network. APs periodically broadcast their presence to nodes within the BSS via beacons. Thus, to join an infrastructure BSS, nodes must receive a beacon.

Released in 1999 as the second 802.11 standard, 802.11b was the first to be widely adopted and most current WiFi devices still maintain backwards compatibility. 802.11b channel bandwidth is 22Mhz, and the minimum transmission power is $0dBm$. This standard introduced High Rate Direct-Sequence Spread Spectrum (DSSS), which increased the maximum data rate from 2Mbps in the original 1997 standard to 11Mbps.

Subsequently in 2003, 802.11g was released, and incorporated into the 802.11 standard in 2007 under clause 19. Using the Orthogonal Frequency Division Multiplexing (OFDM) modulation, transmit rates are provided upto 54Mbps. In deployments where 802.11g

networks must coexist with 802.11b devices, backwards compatibility is supported. Here, data packets are transmitted with an 802.11b-compatible DSSS packet header, therefore reducing the maximum data rate. This revision also incorporated more extensive security features into the standard.

802.11n is the most recent amendment to the 802.11 standard in the 2.4Ghz domain, and is incorporated into the 802.11 standard in 2012, under clause 20. The same OFDM modulation is used as in 802.11g, however, by affording channel bandwidths upto 40Mhz, and support for multiple spacial streams simultaneously, 802.11n can support data rates upto 600Mbps. The larger bandwidth therefore presents a wider interference footprint, which must coexist with other 2.4Ghz devices.

The 802.11 PHY provides an interface to configure the rate of the underlying modulation used to transmit packets. Rate selection algorithms then vary between device vendors. In some cases, 802.11 stations may respond to packet loss by reducing the rate, to improve performance under low SNR. Other instances may respond by increasing the rate, to reduce the on-air time for each packet, and avoid future collisions. A review of 802.11 rate selection algorithms is beyond the scope of this discussion.

Across all revisions, the MAC protocol remains mostly unchanged. 802.11 uses the Distributed Coordination Function (DCF) to mediate channel contention between multiple 802.11 devices. Before being able to use the channel, the protocol requires that the channel is idle for a minimum period called Inter-Frame Space (IFS). Otherwise, the node must enter the backoff procedure, where the node waits until this condition has been met. After which, each node picks a random slot to begin transmitting; the node picking the soonest slot number wins and begins transmitting, while the other nodes repeat the procedure. An Acknowledgement is transmitted after each data frame. This is shown in figure 2.6.

By defining the IFS, nodes can be afforded different priorities, shorter IFS granting higher priority. Short IFS (SIFS) is used for higher priority packets, including acknowledgements, while DCF IFS (DIFS) is used for all other packets.

The DCF requires a CCA method to detect activity on the channel, to determine

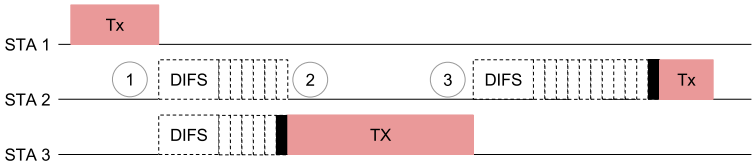


Figure 2.6: Simplified model of 802.11 DCF

if another device may be transmitting; this CCA method affects the coexistence with other neighbouring devices. The standard supports two mechanisms to provide CCA: Energy Detection (ED). which can detect any signal above a preset threshold, and Carrier Sense (CS), which can only detect signals of the same modulation. In 802.11b, g, and n, ED CCA is optional, and is not required in any regulatory domains. However, its implementation has become more common in later compliant devices.

Virtual Carrier Sense (VCS) is provided by the MAC layer using the Network Allocation Vector (NAV). Here, all packets include a duration field, indicating the duration of the packet. Upon detecting a packet, a station marks the channel busy until the duration field has expired. Request-To-Send/Clear-To-Send (RTS/CTS) packets are also defined in the standard; if implemented, the duration field in both records the duration of the entire DATA/ACK exchange. The use of RTS/CTS is configurable, and implementation dependent, and is recommended where improved NAV protection is needed (see hidden node problem).

2.3.3 Microwave Oven (MWO) interference

Microwave Ovens (MWOs) use a magnetron to emit microwave radiation. Despite being shielded by a faraday cage, leaked energy is still emitted across the $2.4Ghz$ frequency range. This is more prominent in older devices. The interference generated is observed as a 50% duty cycle pulse train, whose on/off period is determined by the AC mains frequency. For example, in countries that use $50Hz$ AC mains frequency, the on-duration of microwave interference is approximately $10ms$, followed by $10ms$ off.

Some previous works have observed unique features of MWO interference within 802.15.4 RSSI traces [ZCW+14, BVN+11]. MWO interference has been shown to cause RSSI fluctuations below the noise floor during the on-period, which is not exhibited

by other interference sources. This is attributed to the saturation of the intermediate frequency amplifier chain in the CC2420 radio.

Unlike other interference, MWO interference is not a by-product of wireless communication. Likewise, MWOs have no channel access or arbitration policy, compared to 802.11 for example. Interference is generated regardless of the prior channel state. This must be accounted for in the design of any interference mitigation approach.

2.4 Chapter Summary

In this chapter, the foundations of WSN, MAC protocols, and interference have been discussed. WSN MAC protocols were reviewed in Section 2.2 where CTI has been shown to affect packet loss and energy inefficiency. Typical interference sources in the $2.4Ghz$ frequency domain - common amongst WSN devices - were discussed in Section 2.3. In the next chapter, previous literature on CTI, including mitigation solutions, with respect to 802.15.4 and WSN MAC protocols specifically, is reviewed.

Chapter 3

Related Work

This chapter reviews previous literature related to wireless coexistence and interference, with respect to 802.15.4 and WSN communication. Following an overview in Section 3.1, three research areas are explored.

In Section 3.2, experimental studies of wireless coexistence are discussed, where the effects of interference are measured empirically. Environmental interference is shown to negatively affect 802.15.4 and WSN links, motivating the remaining discussion in this chapter.

In Section 3.3, theoretical models of coexistence are presented, where link performance and energy efficiency are estimated from known environmental properties. Underscoring problem P.1, it is shown here that no model of energy consumption is currently available that accounts for the idle listening behaviour in interference environments.

In Section 3.4, previous solutions to mitigate wireless interference are presented. These are classed into avoidance, detection, and resilience approaches. Exploring detection mechanisms, previous works are shown to either require specialist hardware - beyond the reach of off-the-shelf WSN hardware, or increase idle listening - at the expense of energy efficiency. The review of previous work in this section underscores problem P.2.

Finally, the chapter is concluded in Section 3.5.

3.1 Coexistence Overview

Focusing on 802.15.4 networks, coexistence scenarios with 802.11 are the most commonly researched due to their application overlap. The consensus here is that 802.15.4 links are more susceptible to the effects of 802.11 interference than vice versa. The effects on 802.15.4 networks are twofold. Firstly, collisions and packet loss increase as channel conditions deteriorate, subsequently worsening latency and QoS. Secondly, the mechanisms to mitigate collisions and packet loss - discussed in Section 3.4 - have additional energy demands, which reduce energy efficiency.

The majority of work in this domain has focused on the standard 802.15.4 PHY and MAC layers, or industry standards such as ZigBee. For the specific case of 802.15.4-based WSN MAC protocols however, the effects of CTI on packet loss and energy inefficiency are amplified. Packet loss is compounded by the awkward nature of unicast communication in duty-cycled WSN MAC protocols; latency is further affected due to the infrequent nature of synchronisation and opportunities for retransmission. Low power listening MAC protocols, such as LPL [MHK07] and ContikiMAC [Dun11], are susceptible to false wakeups caused by interference - where CTI is mistaken for WSN channel activity. In the presence of external interference, false wakeups become more frequent, increasing idle listening and reducing energy efficiency.

Yang et al present a thorough survey of 802.11 and 802.15.4 coexistence work in [YXG11]. The authors begin with measurement-based studies, which evaluate experimentally the effect of distance, transmission rate, 802.11 version, and data direction variables on 802.15.4 communication. These typically take place in controlled environments to ensure reliable results. Following this, theoretical studies are surveyed. These are organised based on input, output and behaviour components, which allows for easy comparison between models, and to identify how one model may feed into another. For example, distance and transmission power may be used as input to a path loss function, whose output is received power; this may subsequently be fed into a Bit Error Rate (BER) function to estimate packet loss. Finally, the authors review existing solutions to interference, classed as either inherent - performed permanently during operation, or

on-demand - dynamically employed in response to detecting interference.

The authors review only coexistence amongst standards-based communication, such as ZigBee [All09], Wireless Hart [wir], and ISA100 [isa], but do not include less common WSN MAC protocols. Also, Yang et al only review 802.11 and 802.15.4 coexistence, and do not consider interference from other sources such as microwave ovens and Bluetooth devices. Interference detection mechanisms are summarised to have appeared in literature throughout the network stack, from the PHY through to the MAC layer. These are used to drive on-demand, reactive solutions. The authors find that these detection mechanisms, and the response to their outcome, require further research.

Hayanajneh et al present a survey of Body Area Network (BAN) coexistence issues in [HAUV14]. BANs are a subset of WSN, sharing the same energy efficiency requirements, radio hardware, and communication protocol design, and therefore this survey is relevant to the wider issue of WSN coexistence. The authors consider two types of interference: mutual, originating from inside the same network; and cross-interference, from other sources. In this thesis, only cross-interference is considered. Three wireless standards are discussed as candidates for BAN communication: 802.15.4, 802.15.6, and Low-power WiFi. As with [YXG11], only interference with 802.11 is reviewed, and WSN-specific MAC protocols are not included. The survey considers theoretical, simulation, and testbed studies, leading the authors to make some important observations. Firstly, the coexistence parameters that are evaluated most often vary to suit the type of study. Secondly, most coexistence studies are carried out experimentally, using testbeds with a small number of nodes per network.

In response to concerns governing WiFi and ZigBee coexistence, Thonet et al [TAJC] present a survey of studies into 802.15.4 and 802.11 coexistence, and measure the effect of coexistence on both networks experimentally. This work only reviewed industrial studies, and did not include any academic sources. The findings here that, except under high traffic loads, WiFi does not impede 802.15.4 networks, is contrary to the consensus among work in this field. Further, the ZigBee standard, designed for home automation purposes, is dissimilar to WSN usage, wherein energy efficiency requirements and traffic

patterns differ.

3.2 Experimental studies

This section discusses previous literature measuring the effects of CTI on 802.15.4 and, more specifically, WSN via experimentation. Due to their prevalence outside WSN, the majority of these works have studied only the 802.15.4 PHY and MAC layers. The findings however, are applicable to WSN MAC protocols also. For similar reasons, these works have focused exclusively on the issue of link quality degradation. Experimental studies of energy consumption, which is of less significance to other 802.15.4 applications, has received relatively little attention.

Angrisani et al [ABFS08] measure packet loss in a coexisting 802.15.4 and 802.11b deployment. A WiFi network, consisting of access point and station separated $13m$ apart, is joined by an 802.15.4 network made up of 10 nodes, located near one of the WiFi nodes. By re-locating the 802.15.4 network, and setting direction of data in the WiFi network, symmetric and asymmetric interference scenarios can be generated. Measuring packet loss, the authors consider the effects of coexistence on both networks. The findings support earlier works: 802.15.4 is worse affected than 802.11, and that 802.11 is less able to detect 802.15.4 despite having ED CCA. A tradeoff is apparent in both networks between desired data rate and packet loss rate, which should be decided during the network design phase.

Motivated by the greater signal bandwidth of 802.11n compared to earlier 802.11b/g, Petrova et al [PWMR07] present a measurement study of 802.15.4 coexistence with 802.11g/n. An 802.11 network was deployed to generate controllable interference, which consisted of an access point and station separated by $11m$. Alongside this, an 802.15.4 network, made up of two TelosB nodes spaced $4m$ apart, was placed equidistant from the 802.11 nodes. The spectral separation, physical orientation, and CCA threshold were varied, while the authors measured the link performance of the 802.15.4 network. 802.15.4 packet loss is shown firstly to be dependent on the 802.11 traffic load - higher interferer throughput reduces channel capacity for 802.15.4 traffic. Secondly, greater frequency

separation between both networks improves the SNR of the 802.15.4 receiver, reducing packet loss. Thirdly, 802.15.4 network orientation with respect to the 802.11 interferer is shown to influence packet loss, although this only applies to 802.11n devices equipped with MIMO antennas. Finally, this paper also highlights that the TI CC2420 does not have carrier sense - as suggested by the specification [Ins06] - that would otherwise allow for an improved CCA method.

Hauer et al investigate, and confirm, the detrimental effect of 802.11 interference on 802.15.4 BANs [HHW09]. The BAN used in the paper consists of two 802.15.4 nodes attached to person, separated by approximately $1.5m$. One node continually transmits packets while the receiver measures link quality throughout the experiment, recording the Packet Error Rate (PER). The transmission power of the BAN is used as an independent variable throughout the experiments.

The first experiment takes place in an interference-free environment, generating controlled interference via an 802.11b network. The human subject, wearing the 802.15.4 BAN, walks on a straight path within range of the interferer. On the 802.15.4 channels overlapping with WiFi, link quality is shown to be impaired; PER is shown to be reduced for higher 802.15.4 transmission powers. The authors only evaluate 802.11b interference, whose PHY and MAC layers differ from more modern 802.11 variants. Further, the 802.11 network used was an Ad-hoc network (also known as an IBSS network), which is known to behave differently compared to a BSS network (consisting of Access Point and Station).

In the second experiment, the first experiment is repeated in three urban locations, to measure realistic, uncontrolled interference. In addition to PER, a laptop in the subject's backpack records also 802.11 activity. Here, WiFi activity is shown to be temporally correlated to 802.15.4 link quality, reaffirming the hypothesis that 802.15.4 wireless links are adversely affected by 802.11 interference.

The authors also analyse the availability, and link quality, of 802.15.4 channels from the urban locations. Sha et al conduct a similar analysis in [SHL11b], focusing instead on Home Area Networks (HAN) using 802.15.4 devices. Similar conclusions are drawn in

both papers: firstly, while there is no single channel consistently available which can offer reliable link quality, there is normally at least one channel available, at any given point, that can. This observation is used to justify the use of channel hopping approaches to mitigate interference. In both papers, this is proven with an offline analysis to select the optimal channel for a given time-trace of link quality - demonstrating, with this technique, that loss of connectivity entirely is rare. Sha et al also observe long lasting periods of consecutive packet drops, indicating that retransmissions alone are insufficient to maintain connectivity on an interfered link.

For both BANs and HANs, the design requirements and features differ compared to WSN, in particular the energy consumption demands, and traffic patterns. Nonetheless, these works empirically highlight the effect 802.11 interference, among other sources, has on 802.15.4 links, and reaffirm the importance of effective mitigation methods.

In contrast to earlier works, Liang et al [LPLT10] and Pollin et al [PTH+08] evaluate the interactions, and subsequently the effects of interference, between 802.15.4 and 802.11 on a finer scale.

The initial coarse measurements provided by Liang et al, including a 90-node deployment in a busy lecture hall, thoroughly confirms the detrimental effect of 802.11 interference on 802.15.4, worse so for 802.11b than 802.11g, due to the slower transmission rate and longer on-air time.

Liang et al then configured an 802.11b/g and 802.15.4 network in proximity to each other, each with a single transmitter and receiver. By observing the RSSI trace from a spectral analyser, finer observations are made about the interaction between the two networks. For short distances between the networks, 802.11bg devices are shown to detect and backoff to 802.15.4 transmissions. However, for longer distances, 802.11b/g cannot detect 802.15.4, and can interfere with, and corrupt, such packets. Liang et al identify these two regions as symmetric - where both 802.11 and 802.15.4 can detect each other, and asymmetric, where 802.15.4 can detect 802.11, but not vice versa.

To explain the remaining issue of packet loss for symmetric links, the authors analyse the distribution of bit errors in corrupted 802.15.4 packets. Here, bit errors are localised

near the beginning of the 802.15.4 packets, otherwise being more spread out for asymmetric links. The authors conclude that both WiFi and 802.15.4 detect the channel free, and begin transmitting, simultaneously. The first transmitter to finish - in most cases 802.11 due to the shorter packet durations - then detects the channel busy, and backs off, resulting in the aforementioned bit error distributions in corrupted packets. Based on these observations, Liang et al then describe two interference mitigation methods, Multiple Headers (MH) and Forward Error Correction (FEC), to mitigate regions of symmetric and asymmetric interference respectively. This work is discussed in Section 3.4.2. While their work is based on ZigBee, the implications for 802.15.4-based WSN MAC protocols are identical.

Pollin et al similarly analyse the coarse and fine grained interactions between 802.11b and 802.15.4, finding that, contrary to prior opinion, 802.15.4 does impact 802.11b communication. In a similar controlled experiment, the measured 802.11 throughput is shown to be degraded under heavy 802.15.4 interference, although the impact on 802.15.4 is not reported. Again the authors analyse an RSSI trace taken during the experiment, and conclude that both nodes detect the medium free and initiate simultaneous transmissions. While this work was based on the 802.15.4 PHY, the traffic load measured (up to 45% of channel capacity) is unlikely for WSN applications.

In [BVT+10], boano et al present the first analysis of the effects of 2.4Ghz interference on WSN MAC protocols. The authors study a broad spectrum of MAC protocols, including sender-initiated (X-MAC [BYAH06], LPL [MHK07]), receiver-initiated (LPP [MELT08]) and TDMA-based (CoReDac [VÖ08]). Interference is simulated based on JamLab [BVN+11]: a separate sensor node emits a pseudo-random signal using the test mode of the CC2420 radio. Two interference modes are used - *bursty*, and *semi-periodic* - which are used to simulate most common $2.4Ghz$ interference. Each MAC protocol is subjected to interference, measuring link performance and energy consumption. From this, three traits of MAC protocol design are identified that achieve better performance under interference.

In the first experiment, a designated transmitter node exchanges packets at a fixed

data rate with a receiver node, recording success rate rate and power consumption. A third node simulates interference. The transmission power of the interferer is set above the transmitter and receiver nodes, ensuring that any collisions result in packet loss.

The paper draws attention to X-MAC and LPP, where better packet reception was found if senders retained packets for longer before giving up, thus having more opportunities to receive an initial probe. Abstracting this observation, the authors state that MAC protocols with more frequent initial handshake opportunities are more resilient to interference. Secondly, a variation of LPP is included, whereby receivers broadcast a new probe after receiving a data packet, giving senders the opportunity to send all packets destined to the same receiver in a short time frame (matching the operation of RI-MAC [SGJ08]), which improves both metrics. The authors term this trait packet-train, whereby brief periods of exclusive channel access can be capitalised upon to communicate unimpeded.

The implementation of CCA before transmitting is then investigated, distinguishing between two types of backoff: the first controlling the waiting time of consecutive CCA if the channel is not free, and the waiting time between retransmissions if a collision is detected. Variations of NullMAC - a simple CSMA protocol - are tested under interference conditions. As well as different backoff strategies, a variant without CCA entirely is also included. At the maximum traffic rate, this is found to achieve the best packet success rate, and high energy efficiency. At more realistic WSN packet rates however, quadratic or linear backoffs achieve better latency and energy efficiency.

DCCA is presented in Chapter 5 as a broader component of WSN MAC protocol design to mitigate interference. A simple incorporation of DCCA into ContikiMAC is described in 5 to replace the standard CCA; in this context, the simple policy evaluated in [BVT$^+$10] is close to the testing of the No-CCA NullMAC variant: if detected, interference is ignored. However, the traits presented by boano et al relate to more abstract MAC and network layer protocol design, and are thus compatible with DCCA.

In [BVT$^+$10], energy efficiency is measured as cumulative power consumption throughout the experiment. To evaluate DCCA in Chapter 5, the average radio-on time per

packet received is used instead. Since the radio is the largest source of energy consumption in the communication system, this remains representative of power consumption. The per-packet metric better conveys the cost of communicating in a network, unlike [BVT+10], where the connection between link quality and energy efficiency is less apparent.

3.2.1 Section Summary

This section has discussed previous experimental studies that measure the effect of interference on wireless sensor networks, 802.15.4 PHY and MAC layers. Due to the prevalence of 802.11 networks in many environments over other $2.4Ghz$ interferers, and the overlapping deployment scenarios with 802.15.4, it is the most prominently studied interferer in these works.

Angrisani et al [ABFS08] and Petrova et al [PWMR07] studied the effects of 802.11 b, g and n interference on 802.15.4 links, and found that the latter is worse affected in terms of packet loss. These studies simulated 802.11 CTI in controlled experiments - which may not represent realistic interference conditions and environments. Conversely, Hauer et al complement this by measuring the effects of CTI on 802.15.4 networks in realistic test conditions. The authors similarly found that PRR of 802.15.4 links is impaired by interference.

These studies only coarsely observed CTI. Conversely, Liang et al [LPLT10] and Pollin et al [PTH+08] studied this on a finer scale to investigate the cause of poor coexistence. The authors concluded that, among other factors, asymmetric timing characteristics exaccerbate CTI in such environments.

The studies described above focused only on the 802.15.4 MAC protocol. By contrast, the effects of interference on WSN MAC protocols - which have different requirements - are studied by Boano et al [BVT+10]. The authours found that the design of these MAC protocol exaccerbates the effects of CTI on WSNs.

To the author's knowledge, no experimental studies exist in literature that examine the effect of interference on idle listening, and energy efficiency, in low-power listening protocols. This is due, firstly, to the relative obscurity of these protocols amongst

802.15.4 research in general. This is due, secondly, to the difficultly of measuring energy consumption and node lifetime accurately outside a lab environment. The review of work dicussed in this section confirms the consequences of CTI presented in section 1.1, and justifies the need for further research to mitigate these issues.

3.3 Theoretical Models

The theoretical models presented in this section are able to estimate energy consumption and link performance, as a function of a set of inputs.

To estimate energy consumption in WSN, most models rest on two assumptions. Firstly that, as is the case with most WSN hardware, the radio is the greatest consumer of energy. Secondly, that communication between nodes in the same network - including destructive collisions - constitutes the greatest source of radio usage. Therefore, input parameters to estimate energy consumption range from the physical layer (such as hardware efficiency, transmission power, range) to routing and node deployment. For most of these works, the effect of CTI on energy consumption is not considered.

Conversely, a number of theoretical models of packet loss have been presented in literature that explicitly incorporate CTI. These estimate 802.15.4 packet loss as a function of interferer properties, such as communication power and rate, transmission rate, and distance between networks. By modelling retransmissions also, network latency can also be estimated.

This allows design parameters to be optimised to reach reliability and energy requirements, else evaluate entirely the feasibility of a deployment. For example, most WSN hardware is driven by a finite power source, with little scope for node maintenance and repair. Therefore, the lifetime of the network is a function of each nodes energy consumption, and therefore may be predicted from these models. Theoretical models are also able to provide insight into energy consumption of WSN, in order to design power conservation strategies. Likewise, packet loss in interference environments may be better understood, leading to more effective mitigation mechanisms.

3.3.1 Energy Consumption

Duarte-Melo et al model the expected lifetime of a WSN as a function of data rate, initial energy capacity, and node distribution parameters [DMLM04]. Information capacity is defined as the maximum data that can be transferred in a network, before the first node loses power. The authors equate this model to a linear flow maximisation problem to optimise information capacity, which is shown to be equivalent to maximising network lifetime. The continuous spatial domain of a WSN deployment is divided into grids, and the subsequent discrete model solved using a linear programming approach. Though this model is not empirically evaluated, the authors are able to compare the effects of node placement and routing topologies on network lifetime. For example, the model shows that network lifetime may be extended by more densely positioning nodes closer to the sink, where traffic load is highest. Communication rate is modelled as continuous throughout the network lifetime. The overhead of channel access and idle listening is ignored: packet communication is assumed the primary source of energy consumption.

Wang et al develop a less abstract model of energy consumption of WSN communication, focusing on radio hardware and channel properties as opposed to network deployment patterns [WHY06]. Each radio component is modelled individually, and transmission power is considered variable to suit the communication range. A 1-dimensional spatial model is used to compare the energy consumption of single- vs n-hop routes, assuming negligible channel contention and retransmissions. Unlike [DMLM04], this model includes the communication overhead of a MAC protocol. The authors find that if, on individual links, the minimum transmission power is used that is still able to deliver sufficient SNR single-hop networks are shown to be more energy efficient than multi-hop networks, given stable channel conditions.

Both of these papers model energy consumption as a function of WSN application, topology, or MAC protocol design, and assume ideal channel conditions. Conversely, Alam et al explicitly model the energy cost of adverse communication events - such as collisions and retransmissions - in [ABM+11]. This work complements previous models, in that topology and physical deployment are fixed, while wakeup interval and data rate

are variable. The authors theoretically model the link layer, encompassing error detection and recovery, and the MAC layer, which includes the channel access model, based on a WSN MAC protocol. From this, the probability of *1)* Wakeup and data packet collisions, and *2)* packet errors are derived theoretically, based on channel usage parameters. Then, the energy consumption incurred in each event is measured on typical WSN hardware. Combined with the probability model, energy consumption is predicted. This approach is evaluated in practice, comparing predicted estimations to actual energy consumption to within 8% error.

In both cases, sending and receiving packets is assumed the greatest constituent of energy consumption within the radio. Conversely, the idle listening behaviour is assumed constant, and independent of channel conditions.

Zheng et al model the idle listening in WSN in their evaluation of ZiSense [ZCW+14], accounting for environmental interference. The accuracy of the listening mechanism in BMAC (an LPL protocol) is measured as the false positive/false negative rate. Alongside the interference probability, these are used as input to a function to predict the rate of false wakeups, and the effect on idle listening. Idle listening is shown to be significantly increased due to false wakeups, exacerbated under either high levels of interference, or low traffic conditions. This model suffices to theoretically compare different wakeup mechanisms and sensitivities, however its accuracy is not evaluated empirically.

In order to simplify the models, theoretical estimations rely on network and environmental abstractions. Since these cannot realistically describe a WSN deployment or operating environment, accuracy is inherently reduced [MN14]. Conversely, simulations capture the entire design of a WSN application, only modelling the environmental characteristics.

Avrora is a WSN simulator presented in [TLP05]. Based on an AVR instruction-level emulator, which uses the same cross-compiled program binaries as the actual WSN hardware, this is able to provide fine granularity of energy consumption. Alberola and Pesch extend the avrora simulator to include a model of the CC2420 802.15.4 radio and wireless environment [dPAP08]. This includes all elements of the radio interface:

CCA, RSSI, and LQI, packet transmission and reception. Therefore, Avrora is able to evaluate the energy consumption of protocols directly, without modification. To realistically evaluate an interference environment, a prior recorded RSSI trace, obtained with standard WSN hardware, can be played back during simulation. The effects of this interference can then be measured. The authors show that this is able to accurately predict packet loss compared to an actual deployment. While not evaluated, it would be feasible to also measure energy consumption, and predict the rate of false-wakeups due to interference, with this approach.

To complement models of energy consumption, online estimation techniques are used during deployment to monitor energy usage. This can be used to validate prior estimations. For example, energy use may be reactively reduced to meet lifetime goals [LMMR07]. Dunkels et al. [DOTH07] implement an on-line energy estimation technique in Contiki [DGV04]. Here, the current consumption is measured for each component, in each state, including the radio, sensors and LEDs. As the node operates, the usage of each component is tallied, to estimate total energy consumption.

3.3.2 Packet Loss

Howitt et al [HG03] study the effects of 802.15.4 coexistence on a nearby 802.11b network. The model describes an 802.11 access point and station, the latter being surrounded by clusters of 802.15.4 nodes. Only downlink traffic is considered, and the model assumes neither network can hear the other (i.e. not having ED CCA). The distance between 802.11 nodes varies the SNR of the receiver, and is used to model the size of the effective interference area. Then, the number of 802.15.4 nodes which may cause interference (in the given environment), and the subsequent packet loss of 802.11, is modelled. The authors find that the PER of the WiFi link may be moderated by limiting either 802.15.4 activity, or enforcing frequency separation between networks. Regardless, the authors find that except for close-proximity links, 802.15.4 networks are unlikely to have any impact on 802.11 links.

As the 802.15.4 standard was being ratified in 2007, Shin et al researched the

issue of coexistence with other communication protocols using the same frequency in [SPK07b, SPCK07, SPK07a]. [SPCK07] studies the affect of 802.11 interference on 802.15.4 devices; [SPK07b] extends this to include 802.15.1 interference. In [SPK07a], the affects of interference are considered in both directions: how 802.15.4 networks may interfere with 802.11 networks. Each study was based on the unslotted 802.15.4 MAC protocol, and assumed that transmissions were independent events. Each study produced an theoretical model, subdivided into 1) the probability of a collision, and 2) the affect on BER of a collision; these findings were then validated via simulation. The models accept node distance, frequency offset, and network size inputs.

These works were based on the 802.11b standard. While this is now mostly obsolete, the models are still valid once timing parameters to new standards are incorporated. The authors assumption that neither protocol uses energy detection CCA is not the case with newer deployments, although the model is still valid in scenarios where the link range is beyond the CCA detection range. These models also assume that the 802.11 transmitter is fully saturated: always having a non-empty send queue. Since this is unrealistic in most scenarios, these only present the worst-case scenario of coexistence. The study highlights that the Power Spectral Density (PSD) of 802.11 is not uniform and varies depending on the offset from the centre frequency. Consequently, simulations confirm that the degree of frequency separation between co-located 802.15.4- and 802.11-networks determines the severity of the interference.

Yuan et al present a more thorough analysis of 802.11 and 802.15.4 coexistence in [YWLN13]. The authors first assume that both networks are equipped with energy-detection CCA, therefore defining three coexistence ranges: $R1$: both 802.15.4 and 802.11 are able to detect each other, and hence avoid collisions; $R2$: only 802.15.4 is able to detect 802.11; $R3$: neither 802.15.4 nor 802.11 can detected each other, but 802.15.4 may still suffer the affects of interference. Models for coexistence then estimate link quality as a product of 1) CCA or packet collision, and 2) Bit Error Rate (BER). The model described incorporates MAC Layer retransmissions of 802.15.4, in order to predict throughput and latency. Unlike previous analysis, this work evaluates the models both

experimentally and via simulation.

Yuan et al make a number of key observations via these models. Firstly, the Tx/Rx switching time can have a significant effect upon link quality for short distances between networks, due to 802.11 being able to begin transmitting during this period. Secondly, partial CCA is defined as when a CCA check only partially captures an interference signal, skewing the outcome. This is shown to have a negative effect on coexistence, leading the authors to conclude that a more robust channel sensing mechanism is desired to improve coexistence.

3.3.3 Section Summary

In this section, theoretical models of wireless link performance and energy consumption have been discussed with regard to CTI. Duarte-Melo et al [DMLM04] present a highly abstract model of energy consumption in WSNs, which takes as input node distribution, energy capacity, and data rate. From this, the authors consider how to optimize information capacity and lifetime of a network. This model does not encompass pragmatic aspects of WSN design, such as hardware properties and MAC protocol design. Conversely, Alam et al [ABM+11] model radio hardware, wireless channel properties and network topology, while Alam et al [3] model collisions between nodes. These models are used to evaluate network topology decisions in multi-hop WSNs, in order to optimize network lifetime.

These studies do not incorporate external interference, and derive energy consumption as a function of packet delivery. Therefore, these studies assume that the greatest source of energy consumption stems from radio communication within the WSN. This is not the case with LPL MAC protocols, wherein false wakeups occur as interference is mistaken for valid WSN traffic. Consequently, Zheng et al [ZCW+14] present a model which includes CTI, and measure the effect of idle listening due to false wakeups. However, this model is not empirically evaluated.

Models of packet delivery under interference conditions accept environment, network, and hardware parameters, and include the behaviour of other network's channel use.

Howitt et al [HG03] and Shin et al [SPK07b, SPCK07, SPK07a] represent models of packet delivery for 802.11b and 802.15.4 networks. The authors show that 802.15.4 interference has negligeable effect on 802.11 links, while 802.15.4 packet reception determined by 802.11 data rate, distance, and frequency seperation. Yuan et al similarly present a model which is then evaluated empirically, and highlight a number of observations regarding 802.11 and 802.15.4 coexistence.

As stated in problem P.1, no model currently exists which has been experimentally validated, that is the intersection of these two bodies of work: estimating energy consumption as a function of environmental interference properties, and based specifically on LPL-style MAC protocols. This would be highly useful, as the expected lifetime of a network could be predicted based on prior measurements of an environment - before any nodes are deployed.

3.4 Solutions

CTI solutions seek to either reduce packet loss, or maintain energy efficiency in interference environments. For the most part, these have been mutually exclusive, although some solutions have been evaluated in both contexts [ZCW+14]. In this section, previous studies are classed based on their approach to mitigating CTI which, broadly speaking, may take three forms:

1. **Frequency Avoidance**

   Avoid interference by communicating on another, unaffected channel.

2. **Resilience**

   Make WSN transmissions more resilient to CTI.

3. **Detection**

   Improve the detection mechanisms of interferers, and WSN, so that they may avoid collisions and maintain energy efficient idle listening.

The taxonomy of works covered in this sections is shown in figure 3.1. Within this, the position of DCCA, a detection-based solution presented in Chapter 5, is also shown.

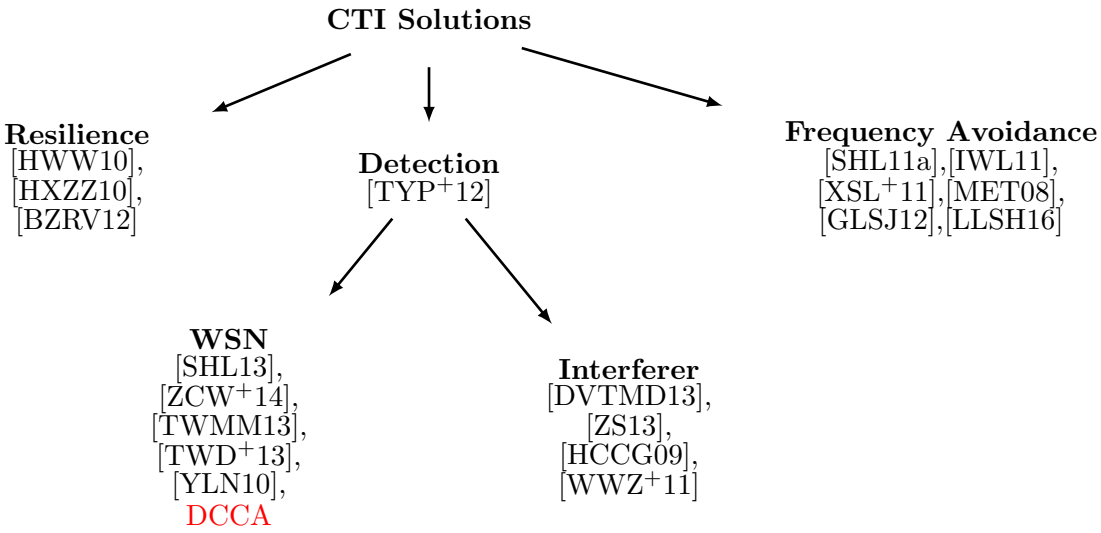


Figure 3.1: CTI Solutions taxonomy

3.4.1 Frequency Avoidance

Frequency avoidance mechanisms mitigate the effects of interference by distancing the WSN from interference sources in the frequency domain. These approaches require that at least a minimum number of unaffected 802.15.4 channels are available at any instant, an assertion supported by previous interference environment studies ([HHW09, SHL11b]). These approaches can be network-centric - all nodes switching to a new channel, or link-centric - applying channel change to affected links only.

In general, these mechanisms share the same components:

- **Detection**

  Detection of deteriorating link quality on a given channel, prompting channel change. To this end, nodes may periodically check channel conditions, else detect deteriorated conditions during the course of operation.

- **Selection**

  Selecting a new channel to use for communication. This may be an empirical process of scanning available channels, or a random selection.

- **Synchronisation**

  Informing other nodes in the network of the new operating channel. In methods

that use a common channel for all nodes, this will end with all nodes performing a channel switch.

- **Communication**

  Communicating on the new channel. In networks without a common channel, this will involve some kind of switching mechanism before each transmission.

Detection is practically interchangeable between these approaches, and may be launched as an integral part of the mechanism, or by a higher layer application. Likewise, the *selection* process is interchangeable: [MET08] and [XSL+11] both use PHY-level measurements based on RSSI, while [IWL11] and [SHL11a] pick the next channel randomly or sequentially with no input.

The synchronisation method is unique in each case, and must be designed to handle:

- Reliable communication of new channel to all affected neighbours.
- A bootstrap procedure, for a node to join an existing network.
- A fallback procedure, in case node synchronisation cannot be achieved on the current channel. For example, due to particularly bad interference.

In [MET08], Musaloiu and Terzis present a frequency avoidance mechanism for multi-hop sensor networks in the context of an environmental monitoring deployment. Atop the ZigBee MAC and routing protocol, periodic data requests are forwarded to a specific node from the base station. On each request, nodes en route conduct scans across all channels to assess their conditions; this data is relayed back to the base station which selects the least-busy channel from this aggregate information. A channel-change request is then sent from the base-station, after which the sensor readings are transmitted.

The authors evaluate packet loss in a multi-hop ZigBee network of four nodes, alongside an 802.11b network which generates controlled interference. Compared to the worst-case single channel - which coincides with the WiFi network, packet loss is reduced from 58% to less than 1%. This method adds significant control traffic overhead, increasing latency and energy consumption even in instances where no channel change is required. Since the

number of control packet increases with the path distance, this centralised approach is unsuitable for large networks. Likewise, this method uses a single channel for all links on a given path, and therefore may suffer in deployments with disjoint interference regions which require more flexible frequency avoidance.

Similar to [MET08], Muzi is a frequency avoidance mechanism which uses PHY measurements, sought from RSSI, to infer channel quality [XSL+11] and inform detection and selection. However, channel selection is per-link, and control is decentralised to each node, thereby improving scalability and reducing overhead. Each node maintains a table of <*node,channel*> pairs for each neighbour, updated whenever a channel change notification is received. To communicate with another node, this table is searched for the correct channel. Nodes periodically measure interference on the current channel; therefore, unlike [MET08] channel scan and selection process is initiated only if the set threshold is exceeded. Following the channel scan, the least-busy channel is selected, weighted in favour of those channels in use by nearby nodes. Notification of the new channel is then unicast individually to neighbour nodes, after which the new channel is switched to.

Muzi is evaluated on a small 4-node testbed in conditions favouring a link-centric frequency-avoidance strategy: two disjoint WiFi stations, covering collectively all available 802.15.4 channels. Two configurations based on fixed channels, which fall under the 802.11 networks, experience 3.5% and 28% packet reception rate; this increases to 94% with Muzi. In the implementation described, the time between the first channel change notification, and the channel change itself is determined by the number of neighbours; Therefore, channel change is not atomic, and may lead to packet loss in networks where nodes have many children. Such a case is not represented by the small evaluation employed. As with [MET08], no fallback option is discussed if control packets cannot be communicated.

ARCH [SHL11a] and Chrysso [IWL11] use higher-level metrics to measure channel performance, namely ETX and recent backoff congestion. These may more accurately represent link performance for a given channel, but inherently take longer to measure and are therefore less reactive to changing conditions. In both cases, channel quality can only

be observed after switching to it, and thus channel selection is either random (ARCH) or sequential (Chrysso). Adjacent channels - which may be affected by the same interferer spanning multiple 802.15.4 channels - are avoided. To reduce the overhead of control packets experienced in [MET08] and Muzi, channel change notifications are appended to outgoing acknowledgements. This increases the time taken to change channel, and amplifies the atomicity issue - which ARCH addresses.

ARCH is designed for Home-Area Sensor Networks (HANs). A channel change is prompted once conditions deteriorate below the threshold. Once a channel is found to be suffering under interference conditions, it is temporarily blacklisted from future selection; once too few channels become available, the blacklist is cleared. While applicable in HANs with relatively stable interference, this approach may be inefficient in dynamic environments with constantly changing interference.

ARCH is firstly evaluated offline based on packet delivery traces, obtained from ten sensor network deployments in an apartment building ([SHL11b]). These deployments are subject to uncontrolled interference originating from microwave ovens, WiFi, Bluetooth devices of the occupants, and thus [SHL11a] gives the most realistic evaluation. In each apartment, the nodes cycled through the available 802.15.4 channels, recording packet error rate in 5 minute intervals - the experiment lasting 24 hours. The findings show that the selection algorithm outperforms random, and fixed selection, and is only 6% below the optimum achievable packet delivery. Also, ARCH requires relatively few - at most 25 - channel switches per day to avoid interference. ARCH is then evaluated in practice using the same testbed. In both single-hop and multi-hop configurations, ARCH increases packet delivery rate, and reaffirms the earlier findings.

Chrysso incorporates frequency avoidance into the routing protocol, although channel change is still coordinated per-link. Nodes append channel quality information to outgoing data packets sent to the parent node. There, the aggregate of this information is used to determine if a channel change is required. Channels are switched, sequentially avoiding adjacent channels which may fall under the same interferer. The implementation described in [IWL11] uses only a subset of five 802.15.4 channels. As with ARCH, channel

change is communicated via ACK packets, however the authors also describe a robust bootstrapping procedure to join a network, and include logic to re-synchronise if control packets cannot be exchanged.

In addition to packet loss, [IWL11] measure energy consumption, thereby evaluating the energy cost of frequency avoidance strategy. Chrysso is tested on two sensor network testbeds under interference conditions. Compared to a fixed channel, Chrysso reduces packet loss and energy consumption; more so under heavy interference. This indicates that the energy cost of overhead traffic is outweighed by the improved channel conditions.

These approaches represent single channel, *reactive*, approaches - in that channel change is instigated in response to changes in channel quality, such as increase in channel activity or packet loss. Recent multi-channel MAC protocols on the other hand, define a multi-channel *proactive* approach - which hop between channels systematically during packet transmission as part of the MAC protocol design [KT13, GLSJ12, MGC16], similar to Bluetooth.

Oppcast is a receiver-initiated MAC protocol [MGC16] that combines opportunistic routing and multi-channel communication, providing spacial and spectral diversity respectively to counter local interference sources. Receivers periodically broadcast probes on a subset of 802.15.4 channels; to transmit, nodes listen on each channel for a probe, before transmitting the packet. *Fast Channel Hop* is defined as an efficient rendezvous mechanism, where receivers and transmitters cycle channels symmetrically until a probe is received. Instead of a stringent routing path, nodes forward the packet on the next hop closer toward the sink, inherently avoiding interfered links. Oppcast is evaluated against single channel-opportunistic, and multi-channel tree-based routing protocols. In a 96-node testbed, Oppcast achieves increased packet delivery, lower latency, and lower energy consumption under interference conditions. In ideal channel conditions, however, Oppcast experiences increased latency compared to single-channel protocols - a cost of additional channel rendezvous. Therefore, reactive approaches may achieve better performance in cases where interference is less dynamic, and can be responded to quickly, while proactive approaches such as this may better suit dynamic environments which

mandate frequent channel changes.

[LLSH16] describes ART, a frequency planning technique that assigns each node's frequency based on its location within the network and measured environmental interference, including WiFi. The frequency domain is divided into a continuous domain (within hardware limitations), rather than the $5Mhz$ increments defined by the standard [IEE07]. Thus, nodes in close proximity are assigned isolated frequencies far apart, otherwise using nearer frequencies. This approach gives the WSN more freedom to avoid WiFi in the frequency domain, yet contravenes the standard. The authors describe centralised and distributed implementations, the latter requiring significant overhead to coordinate channel assignments. Each node must also know its location within the network, which is unlikely in many deployments.

A common feature of all these works is evaluating link conditions, in order to inform channel selection. This shares many similarities with Link Quality Estimation (LQE): metrics which are used in route selection to structure the network. For example, PRR and RSSI metrics may be used to determine the most suitable next hop in a network, for a given node. Given the variety of LQE goals (such as reactivity, stability, and accuracy), approaches, and evaluation metrics, there has been a large body of work investigating LQE.

Baccour et al survey WSN link studies and LQE methods in [BKM$^+$12], separating methods into *hardware* - such as RSSI, SNR, and *software* - such as PRR, ETX. LQE's operate on a per-link basis, and therefore can only extract information from delivered packets between nodes. Conversely, frequency avoidance methods require measurements per-channel, although some may aggregate per-link information ([IWL11]). Baccour et al conclude that combinations of LQEs, hardware and software, may be more accurate than a homogeneous metric [BZV$^+$10]. Similarly frequency avoidance mechanisms which combine hardware and software measurements could allow for more efficient channel selection; this remains an open research question.

These works have shown that frequency avoidance is a viable option to mitigate interference: by switching to an unaffected channel, the adverse effects of interference can

be avoided entirely. Link metrics, including packet loss rate and retransmission count, are reduced where frequency agility is employed. In all cases, additional control overhead is required, such as to aggregate channel readings and initiate channel change, which may incur additional energy cost. Chrysso [IWL11] found, however, that the energy cost is outweighed by the benefits of frequency avoidance; for example, by requiring fewer retransmissions.

Studies measuring interference in typical environments have found that, in most cases, interference avoidance is sufficient to mitigate interference. However, channel availability is dynamic, leaving no single channel consistently available. Since Chrysso and ARCH describe only limited fallback options, relying on some other channel being deterministically available to rendezvous, it is feasible that a network make become disconnected despite other channels being unaffected. Therefore, other solutions that are able mitigate interference on the same 802.15.4 channel are justified in order to complement these frequency avoidance approaches. The issue of switching atomicity is mitigated only by ARCH, however none of these methods have been evaluated under such conditions - where any nodes have more than one child node. Similarly, these solutions must ensure agreement between nodes during channel selection and synchronisation marred by CTI, else channel change may not be unanimous [BZRV12].

Scope for future work in this domain includes a comparison of software-based, NET-layer measurements (such as ETX, congestion) vs hardware PHY-layer metrics (RSSI) for channel selection. An approach which incorporates both of these approaches could quickly eliminate bad channels via coarse PHY measurements, then more closely scrutinise channel selection based on finer NET measurements. From these methods, a distributed scheme is preferred over a centralised approach, but may lead to unnecessary fragmentation of the network over multiple channels. An approach which fragments the network as little as possible while still improving link-level reliability would build on these works.

3.4.2 Resilience

Resilience approaches mitigate the effects of interference - particularly packet loss - by making packet transactions more resilient to CTI. These mostly reside at the link layer, and are integrated into the MAC protocol. The standard 802.15.4 MAC Automatic Repeat Request (ARQ) scheme is a simple example: unacknowledged packets - the result of either data or ACK frame being corrupted - are retransmitted after a backoff period. Hence, the recipient is given another opportunity to receive the packet. Other methods discussed in this section strive for more efficient post-collision recovery mechanisms, while others may embedded additional redundancy in transmitted frames to permit error correction.

Liang et al [LPLT10] focus on CTI between WiFi and ZigBee networks, and identify two interference regions: *1)* symmetric - both WiFi and ZigBee can hear and avoid each other, and *2)* asymmetric - WiFi cannot hear ZigBee, but collisions still result in packet loss. Following this, Multiple Headers (MH) and FEC are proposed to counter these interference regions respectively.

MH is based on the observation that while both networks may be able to detect the other, simultaneous channel access may still result in a collision. Therefore, WiFi packets are more likely to collide with the beginning of (much longer) ZigBee packets - including the preamble and header which are required to detect and receive incoming frames. MH encapsulates multiple repeated packet preamble and headers within transmitted ZigBee frames, giving receivers multiple opportunities to detect incoming packets. MH is evaluated in a symmetric interference environment of five 802.11g clients and access point, and five ZigBee nodes. Across all links, PRR is shown to be improved 50% with only a single additional header. Further headers provide diminishing improvement. Jamieson and Balakrishnan similarly mitigate preamble and header corruption via a postamble at the end of each packet [JB07], which can also be used to detect incoming packets stored in a circular buffer. This approach adds significant hardware complexity, and is beyond typical WSN radio design.

FEC is used to embedded additional redundancy, so that receivers may recover

corrupted packets. Two FEC implementations are evaluated based on collected packet traces: Hamming codes and Reed-Solomon. Due to the localised nature of decoding errors, Hamming codes were only able to correct 4.7% and 19.1% of 802.11b and 802.11g interference respectively. On the other hand, Reed-Solomon is able to correct 85.9% and 85.3% of corrupted packets respectively. Liang et al present TinyRS, a Reed-Solomon implementation optimised for embedded devices; however, this does still consume significant resources: $2.9KB$ ROM, $1.4KB$ RAM, and on the Tmote Sky platform requires at least $207ms$ to decode corrupt packets. Comparing against ARQ and Partial Packet Recovery (PPR), the authors find that FEC is more efficient under heavy interference. However, in lesser interference environments, ARQ/PPR schemes may prove more efficient.

From this, Liang et al present BuzzBuzz: a resilience approach that reactively employs ARQ (initially), MH, and FEC as interference worsens. Evaluated on a large WSN testbed in an office environment, BuzzBuzz is found to improve packet delivery from 43% to 73%, while also reducing traffic load on the network. Since the maximum frame size in 802.15.4 is fixed, the payload overhead incurred by MH and FEC curtails the data capacity of each packet.

In [HXZZ10], the distribution of idle period lengths within WiFi traffic is modelled as a Pareto probability distribution. The authors find that despite there being sufficient channel capacity for ZigBee, the CSMA MAC is unable to mitigate the effects of collisions. Intuitively, smaller packets have less on-air-time, and are less likely to collide with other interference. Therefore, large packets may be made more resilient to bursty interference by dividing them into smaller packets, the sum of which is more likely to be successfully transmitted. This incurs the cost of additional overhead, as each packet requires it's own PHY/MAC header and footer.

WISE is proposed as a MAC solution which transparently segments network traffic into smaller packets. The maximum packet size is optimised to achieve throughput goals in a measured interference environment. In their evaluation, WISE achieves higher PRR and throughput, but also reduces the overhead of the network - incurred otherwise via more retransmissions.

Boano et al address the more narrow issue of wireless agreement, a fundamental component of many WSN protocols, in interference conditions [BZRV12]. Many protocols throughout the network stack require agreement between nodes, including frequency hopping and TDMA protocols. Known as the *Two Generals' Problem*, in an n-way handshake facilitating agreement it is impossible for the final packet to be conclusively sent. Disagreement occurs when this packet packet is lost, and both nodes differ on the perceived outcome. To reduce the chances of disagreement, the final packet must be made more resilient.

A simple solution is k-MAG, where the last packet is repeated k times, ensuring a higher likelihood of at least one being received. This incurs the additional overhead of each transmission. Boano et al instead propose JAG, which jams the channel using the test mode available on some 802.15.4 radio, as final exchange in a handshake. This is detected via an RSSI trace, which can easily detect the absence of any jamming signal. Provided the duration of the jamming signal exceeds the maximum duration of any other interference, it can be reliably detected. JAG is evaluated firstly in a 15-node testbed under controlled WiFi and Bluetooth interference, and secondly in a residential deployment subject to environmental interference. Compared to 2-MAG, JAG achieves fewer disagreements while also reducing the required listening duration. Microwave oven interference is not included in the evaluation. However, since this typically has a period greater than the listen threshold in JAG, the detection algorithm may be liable to *false-positives* under such conditions, increasing the rate of disagreements.

Unlike these pre-emptive resilience approaches, Hauer et al describe a reactive approach [HWW10]. After a collision has occurred, receivers respond with a Negative-Acknowledgement (NACK), indicating which parts of the packet are unreadable. The sender then retransmits only the required parts of the packet.

How to identify corrupted packet segments, while minimising overhead, has been considered as well in previous literature. Jamieson and Balakrishnan propose softPHY, an augmented PHY interface which annotates the confidence of each symbol on receiving a packet [JB07]. This approach is not yet available on most commodity radio hardware,

including 802.15.4. Hauer et al instead propose REPE to solve this problem, which samples the RSSI of the radio *during* packet reception. Within this trace, external interference is observed as spikes above the flat 802.15.4 packet profile. These are then temporally correlated to byte-locations within the payload most likely to be incorrect. REPE is evaluated on two Tmote Sky testbeds. Compared to the default ARQ-scheme, REPE achieves 2.1% and 6% throughput gain.

3.4.3 Detection

In deployments where frequency avoidance is not possible, improving the detection of heterogeneous network technologies can ensure fair channel usage, and mitigate the effects of CTI. Homogeneous detection is an existing component in wireless protocol design, necessary to reduce collisions in contentious channel conditions. For example, 802.11 must listen to the channel before transmitting, to reduce the probability of multiple nodes interfering simultaneously. Such protocols often cater for heterogeneous networks also, however, differences in their implementation prevent fair channel access. Prior work has found this to be the case with 802.15.4 and 802.11 networks: the latter is unable to detect the former, leading to collisions and preventing efficient channel arbitration.

Detection mechanisms also play an important role in LPL-style MAC protocols in WSN. Here, the channel state is used to infer if another node is transmitting, to decide if the radio should wake up or conserve energy by returning to sleep. However, this mechanism can be unintentionally triggered by interference on the same channel, leading to a *false wakeup*. This reduces energy efficiency and, in battery-powered WSN, network lifetime. From this, a logical solution is therefore to bolster these detection mechanisms' sensitivity to heterogeneous devices. If the WiFi MAC protocol, for example, could detect 802.15.4 signals, transmissions would be deferred until the channel becomes free again, avoiding a collisions.

In the following discussion, previous literature is classed based upon where detection is improved - either the interferer, or the WSN. Most interferer solutions have almost exclusively focused on 802.11. This classification does not dictate where any modification

takes place. For example, making WSN transmissions more visible to 802.11 is considered an interferer-detection mechanism, even if the modification resides on the WSN.

Tytgat et al study the asymmetry in ZigBee and WiFi CCA mechanisms in [TYP+12]. The effect of WiFi interference on ZigBee is modelled, on the assumption that WiFi CCA cannot detect ZigBee transmissions. As with carrier sense CCA in WiFi, Tytgat et al also find it to be the case even with energy detection CCA. This is due to the difference in bandwidths. 802.15.4, having a bandwidth of $2Mhz$, has 9.6-dBm higher sensitivity to WiFi, that has a bandwidth of $20Mhz$, than vice-versa.

Coexistence Aware CCA (CACCA) is then proposed to supplement WiFi and ZigBee CCA with a sensing engine, specifically to detect each others signals. The CACCA duration is modelled around 802.11g CCA of $4\mu s$. No signal processing is done by the sensing engine, however the authors assume that sufficient accuracy is possible by measuring signal strength alone. The prior model is then adapted to include ZigBee-only CACCA, WiFi-only CACCA, and both.

Under $100kb/s$ WiFi interference, the model then predicts that WiFi-only CACCA may reduce ZigBee packet loss by 75%. Conversely, ZigBee-only CACCA results in 24% reduction, while both results in 99.6%. WiFi-only CACCA is also shown to decouple the WiFi traffic load from ZigBee packet loss, otherwise correlated in the original model. Tytgat et al here only model the timing interaction of WiFi and ZigBee, with and without CACCA. The sensitivity of CACCA is not modelled which prevents analysis of the transmission power asymmetry between WiFi and ZigBee. Nonetheless, their work affirms that WiFi is the primary culprit in WiFi-ZigBee CTI. Following, in the case of 802.11, interferer-side solutions are likely to yield the greatest reduction of 802.15.4 packet loss in joint deployments.

3.4.3.1 Interferer-side

Valck et al present an implementation of WiFi-only CACCA on an Software Defined Radio (SDR) platform in [DVTMD13]. The authors design CACCA to be backward-compatible with the existing 802.11 standard, constraining the CACCA duration to $4\mu s$.

Two methods are studied to implement CACCA: energy detection (ED), and matched filter (MF). ED measures the energy on each overlapping 802.15.4 channel, and MF also filters each channel for the O-QPSK modulation.

CACCA is then implemented on a WARP SDR platform. To evaluate CACCA, the 802.11 MAC protocol is mimicked in order to simulate coexistence. For the evaluation, a shielded RF enclosure is used, and the signal attenuation is manually adjusted between the WiFi interferer, and two ZigBee nodes. Measuring CACCA sensitivity, ED is shown to retain 90% accuracy for signals above -79dBm, while MF reduces this to -83dBm. By comparison, the 802.15.4 standard requires a sensitivity of -85dBm. However, the authors explain that this sensitivity is not possible for ED and MF CACCA within the CCA timing constraint of the 802.15.4 standard.

The second experiment includes an 802.11 interferer, with and without CACCA, alongside two ZigBee nodes transmitting at maximum capacity. Within those sensitivity ranges, CACCA is shown to limit packet loss to less than 10%, wherein otherwise total packet loss is experienced. This evaluation compared against an 802.11 benchmark that did not have ED CCA. More useful would have been to measure against an ED CCA-equipped WiFi interferer, which is supported by the standard. This approach requires extensive modification to the design of WiFi hardware. Given the guarded nature and complexity of 802.11 firmware; the number and ubiquity of WiFi networks already deployed, this approach cannot practically alleviate CTI in the immediate future.

By contrast, Hou et al [HCCG09] and Tang et al [WWZ+11] modify WSN devices: leveraging the existing detection mechanism in WiFi, improving the visibility of 802.15.4 communications. In both cases, this is achieved via the sink node, which is responsible for signalling to nearby WiFi devices during ZigBee packet exchanges. Hou et al build on the 802.11 VCS MAC component, which uses RTS/CTS packets to reserve the channel for the duration of the packet exchange. To utilise this for the protection of 802.15.4 packets, an 802.11 CTS frame is broadcast by the sink, listing a duration sufficient for the following ZigBee exchange. Upon hearing this CTS frame, WiFi devices will defer until the channel becomes free again after the ZigBee exchange.

Similarly, Wang et al propose WiCop, and study two signalling approaches: Fake-PHY headers, and DSSS-nulling. The former broadcasts 802.11 packet headers, whose length field is sufficient to cover the ZigBee exchange. This is similar to the VCS mechanism described above, but is a more explicit deviation from the 802.11 standard. DSSS-nulling instead continuously transmits 802.11 PHY headers, jamming the channel, throughout the ZigBee packet exchange. The signal bandwidth is reduced from $22Mhz$ to $8Mhz$, in order to provide sufficient space on the channel for ZigBee, while still remaining visible to WiFi interferers. This approach guards against opportunistic WiFi devices that may ignore Fake-Headers and CTS approaches.

In both papers, these methods are found to be capable of reserving sufficient channel capacity to meet the QoS requirements of the WSN. Evaluated under WiFi interference, packet loss is reduced to below 3% and 2% respectively in [HCCG09] and WiCop [WWZ+11]. Both papers envisage a medical sensing application, and assume a centralised topology around the sink node.

These approaches require atypical hardware solutions, beyond the remit of off-the-shelf WSN. To avoid this pitfall, Zhang and Shin present a signalling approach based on standard 802.15.4 hardware. During the ZigBee packet exchange between two nodes, a third node - designated the signaller - emits a busy tone on an adjacent channel. The signaller may have a greater transmission power than standard ZigBee nodes, but otherwise conforms to the design of 802.15.4 hardware, and is detectable by WiFi devices. The busy tone covers the CCA check, packet transmission, and acknowledgement of the ZigBee CSMA protocol. Therefore, the issues of asymmetric interference range and Tx/Rx switch found in [LPLT10], are mitigated. This approach is termed Cooperative Carrier Signalling (CCS), and is evaluated on the MICAz and USRP SDR platforms [ZS13].

In CCS, packet transmissions are preceded by a RTS/CTS exchange which includes the packet length. Upon overhearing this exchange, the signaller switches to an adjacent channel - still under the same WiFi channel, and emits the busy tone. Therefore, WiFi nodes are able to detect and avoid ongoing ZigBee communication. CCS is evaluated in

a testbed which includes co-located ZigBee and WiFi networks. On a single link, CCS is shown to reduce ZigBee packet loss by upto 90%.

CCS is also evaluated for ZigBee in TDMA mode, where similar improvements are shown. This approach does not require any hardware modifications outside either standard, and the authors describe more generally how it may alleviate coexistence between arbitrary networks. The overhead of coordinating with the signaller does incur a cost in energy consumption of at least 10% above standard ZigBee. This is detrimental in WSN applications where energy efficiency is paramount. Further, Zhang and Shin recommend the signaller be mains powered and capable of higher transmission power - not possible in many WSN applications.

3.4.3.2 WSN-side

WSN-side detection solutions mitigate CTI by affording the presence of heterogeneous interference in the design of those protocols. These solutions are designed to reduce either packet loss or false wakeups - the latter is only applicable to LPL MAC protocols. For example, packet loss may be reduced if WSN nodes are able to handle collisions from WiFi interference, or within the same network, differently [TWMM13]. Likewise, LPL protocols may be made more efficient under interference if the rendezvous mechanism were able to differentiate WSN traffic from interference, waking up only for the former [SHL13, ZCW+14].

Yuan et al and Tang et al both present interference mitigation mechanisms based on CCA threshold adaptation [YLN10, TWD+13], which aim to reduce ZigBee packet loss in interference environment. Under Wifi interference, Tang et al show that the CCA threshold of ZigBee devices heavily influences the rate, and cause, of packet loss. Increasing the threshold reduces the rate of CCA collisions, and subsequently is able to achieve higher packet delivery. However, if the threshold is set too high, collisions with WiFi interference may increase. Therefore, a equilibrium must be found. Both approaches adaptively change the CCA threshold: Yuan et al compare the rate of CCA failures to a preset threshold, while Tang et al measure the rate of packet buffer overflows. The former

approach is evaluated in a network simulator comprising a WiFi network and sixteen ZigBee nodes; the latter is evaluated on a small-scale testbed. Both evaluations show that packet loss, caused by CCA failure and collisions with interference, is reduced. However, neither evaluates the affect of this CCA threshold adaption on communication within the WSN, where multiple nodes may be vying for channel access simultaneously. These methods are based on ZigBee CSMA, which has only one CCA check before transmitting. This is vastly different to WSN MAC protocols which have multiple packet strobes per transmission, each preceded by CCA.

Tang et al present Interference Aware Adaptive Clear Channel Assessment (IAACCA), which more proactively contends for channel access by replacing the standard CCA [TWMM13]. Instead of a single CCA check, the channel is sampled continuously until found to be clear. If a timeout is reached, the packet is dropped. Otherwise n further CCA checks are taken - where n is random to avoid collisions with other ZigBee devices - after which, the packet is transmitted. Compared to the standard CSMA/CCA mechanism, which more conservatively backs off on finding a busy channel, IAACCA is shown to reduce packet loss under WiFi interference. IAACCA is evaluated only in a small network, where any interference is assured to have originated outside the network. In a heterogeneous environment however, a collision policy informed by the interferer source would be more beneficial: able to enact the most suitable response deterministically. Nonetheless, this approach demonstrates that the standard CSMA/CCA mechanism, and its sweeping CCA collision response, is inadequate in CTI environments (particularly under WiFi interference). DCCA is evaluated in this context in Chapter 5, and found to reduce packet loss by employing such a policy.

To mitigate false wakeups incurred by LPL MAC protocols, Sha et al present Adaptive Energy Detection Protocol (AEDP) [SHL13]. In 802.15.4 compliant radios the Energy-CCA threshold, above which any signal indicates a busy channel, is configurable in software. The authors find that this parameter affects the rate of false wakeups: higher thresholds are less liable to detect other interference and cause false wakeups; it must be low enough, however, to detect valid incoming packets. The CCA threshold must

therefore be raised as much as possible, but not so much as to miss valid packets. In order to adapt to different environments, AEDP seeks this optimisation at runtime.

Each node tracks two variables within a sliding, 15 minute window: *1)* ETX from incoming packets (ETX), *2)* false wakeup rate (WR). For both variables, upper thresholds are set to guide the refinement algorithm: if $ETX > ETX_{threshold}$, the CCA threshold is quickly reduced; if $WR > WR_{threshold}$, the CCA threshold is gradually increased. Over time, AEDP will find the most efficient CCA threshold which balances ETX and false wakeups. AEDP must be able to adapt to changing network topology, which may require lowering the CCA threshold to detect new neighbours. This is achieved by periodically setting the CCA threshold to its lowest value. Since this is neither energy-efficient or highly reactive, AEDP is better suited to stable network topologies.

AEDP is implemented in BoX-MAC-2 [ML], using the default wakeup interval of two seconds. The node duty cycle is used to measure energy efficiency with, and without AEDP - using the default CCA threshold. In a quiet environment, false wakeups are rare, and hence AEDP brings no changes in duty cycle. Alongside an 802.11n network generating interference, AEDP reduces duty cycle by 47.3%. Likewise, in a residential deployment, AEDP is able to reduce the duty cycle by 45.5%. AEDP is tested in a multi-hop network on a 55-node deployment in a residential deployment exposed to interference, each node sending a packet to the sink every five minutes. During the 24-hour experiment, the duty cycle is again reduced to 35.44% using AEDP. Interestingly, AEDP reduces the average ETX by 11.26%, due to unintentionally pruning unstable links which are below the CCA threshold. Since AEDP cannot reduce the CCA threshold below the lowest neighbours RSS, this is shown to determine the effectiveness of AEDP. Thus, the authors note that AEDP may offer no benefit in sparse deployments.

Zheng et al point out that many deployments are dominated by such Intermediate Quality (IQ) links, with RSS indiscernible from other interference sources. Examining a dataset which includes link RSS of a large testbed, 90% of links are shown to be below -66dBm. Following this, Zheng et al describe ZiSense - a mechanism to reduce false wakeups not reliant on signal strength [ZCW+14]. ZiSense is similar to other interference

classification mechanisms [ZXX+10, AAM11], insofar as an RSSI trace is searched for known spectral and temporal features. However, ZiSense is intended only to detect the presence of 802.15.4, in order to bolster the rendezvous mechanism.

Instead of a single CCA check, ZiSense samples the RSSI register at high frequency over a sample period. From the RSS trace, ZiSense identifies individual signals, and for each constructs a feature set consisting of 1) On-air time: the duration of each signal, 2) Peak-to-Average-Power Ratio (PAPR): a measure of the shape of a signal, 3) Minimum Packet Interval (MPI): the minimum interval between successive transmissions, and Under Noise Floor (UNF): a binary indication of RSS dips beneath the noise floor, characteristic of microwave oven interference. (1) and (3) can be correlated for each interference source, for example the maximum value of (1) for an 802.11g device is $542\mu s$. The authors found that PAPR differs depending on the modulation technique: WiFi has a higher value than Bluetooth/ZigBee due to different modulation techniques. Each feature set is then passed to an identification algorithm, three of which are described in [ZCW+14].

The false wakeup rate of ZiSense is first evaluated under controlled conditions, compared to B-MAC and AEDP under WiFi, Bluetooth, and microwave oven interference. B-MAC, which uses a fixed CCA threshold, performs the worst, approaching 100% false wakeup rate for short distances. AEDP reduces this only when the interferer is further than $4m$ away, or when the 802.15.4 links have RSS greater than -$65dBm$. This is expected, since high interferer, or low 802.15.4, signal strength render both indistinguishable. On the other hand, ZiSense consistently achieves fewer false wakeups independent of link RSS or interferer distance than AEDP or B-MAC. ZiSense is then evaluated in a large-scale testbed of 41 nodes in an office environment. Each node in the multi-hop network forwards one packet every five minutes to the sink, and the channel check rate is $2Hz$. ZiSense is shown to reduce the duty cycle from 3.74% (B-MAC) and 4.14% (AEDP) to 2.46%. In the ZiSense implementation, the duration of the RSSI sampling is $2800\mu s$ - 90 samples. This is a significant departure from a single CCA check ($128\mu s$), and adds significantly to the cost of idle listening.

P-DCCA is presented in Chapter 5. As with ZiSense, P-DCCA searches an RSSI trace for prior known temporal and spectral features, in order to differentiate 802.15.4 from other interference. However, to shorten the timescale on which such features can be detected, P-DCCA does not rely on inherent 802.15.4 characteristics. Instead, P-DCCA nodes embed additional information in the amplitude of outgoing transmissions - by varying the output transmission power, effectively creating an orthogonal channel. Detection of this unique feature in an RSSI trace indicates an P-DCCA transmission. The duration of a P-DCCA check is variable between $128\mu s$ and $256\mu s$, more closely approximating that of a standard CCA check. In Section 5.5, P-DCCA is shown to significantly reduce the cost of idle listening compared to ZiSense.

As with these approaches, P-DCCA is designed to mitigate the false wakeup problem in low power MAC protocols, however, P-DCCA is also designed to alleviate channel contention and packet loss in interference environments. P-DCCA and ZiSense search for signal features discernible from interference, that are invariable with distance or signal strength. Consequently, in either a diverse link environment, or one subject to link dynamicity, both would expect a lower false positive rate compared to AEDP. However, the underlying CCA in AEDP relies on a simple RSSI threshold, and is inherently less prone to missing valid signals, and therefore may suffer fewer false-negatives. The drawback of P-DCCA is that the power variation used to identify transmissions reduces the total SNR of each packet, and hence may reduce the communicable link range. As with AEDP, P-DCCA may therefore be less suitable in sparse deployments, although such links are more heavily penalised in P-DCCA.

P-DCCA is evaluated against ZiSense in Chapter 5, measuring true-positive and true-negative accuracy. While the high accuracy recorded in [ZCW+14] could not be replicated, P-DCCA was shown nonetheless to achieve comparable or greater accuracy. The authors note that interference sources other than those considered in [ZCW+14] may be less discernible from 802.15.4, yielding lower true-positive accuracy. In Section 5.4, this is found to be the case with 802.11b interference, under which ZiSense is highly susceptible to interference. While now mostly obsolete, there are circumstances where

802.11b interference may be encountered, where ZiSense would perform poorly. This is due to the modulation and timing similarities between 802.11b and 802.15.4.

Both P-DCCA and ZiSense rely on preset thresholds to differentiate unknown interference from incoming traffic. These face a similar tradeoff to the CCA threshold, and therefore a mechanism based on AEDP may be used to fine tune such parameters at runtime - in order to optimise the false-negative/positive tradeoff.

Detection has been proven an efficient solution to reduce packet loss and maintain energy efficiency in CTI environments. As the most prolific example of CTI, studies here have mostly focused on 802.11 and 802.15.4 coexistence. Improving the ability of WiFi to detect other networks is shown to yield the greatest improvement in 802.15.4 packet delivery. This is because WiFi devices are unlikely to detect other CTI, and whose transmissions are most destructive. Solutions here based on modified radio hardware - to improve either sensitivity of, or visibility too, WiFi CCA, achieve the greatest improvement in 802.15.4 link performance. However, the cost and energy consumption of these additional components precludes many WSN applications.

In WSN-side detection solutions, packet loss is mitigated by affording more selective channel arbitration. Collisions with other WSN traffic are avoided, while more assertively vying for channel access amongst interference. WSN-side detection solutions are also able to ensure energy efficiency in CTI environments, by avoiding false wakeups. In Chapter 5, P-DCCA is presented as a WSN-side detection solution. P-DCCA has greater accuracy compared to existing approaches with minimal energy overhead, and is shown to both mitigate packet loss, and preserve energy efficiency.

3.4.4 Section Summary

Among coexistence solutions in literature, the three most prominent classifications discussed are frequency avoidance, detection, and resilience. Each solution has merits and tradeoffs that suit particular circumstances; these must be accounted for when considering which to employ in a given environment.

Across all solutions, comparison of the results suggests that - annecdotally - frequency

avoidance offers the greatest improvement in link quality; this is because interfered channels can be avoided entirely allowing unimpeded WSN operation. While reducing packet loss, this will also reduce false wakeups and improve energy efficiency - although the latter has not been evaluated in practice. Musaloiu and Terzis [MET08], Sha et al [SHL11a], and Iyer et al [IWL11] present frequency avoidance solutions that select the channel based on channel measurements, and network metrics. These are reactive approaches - which initiate channel change in response to adverse channel conditions, and incur the cost of channel coordination across multiple nodes in a WSN. By contrast, Kumar et al [KT13], Gonga et al [GLSJ12], and Mohammad et al [MGC16] present multi-channel reactive approaches, whereby WSN nodes continuously iterate through a subset of channels as part of the MAC protocol.

Frequency avoidance requires that a minimum number of channels are not subject to interference at any instant. While previous studies of BANs and HANs support this assumption [HHW09, SHL11b], the emergence of new, wide-bandwidth, standards may challenge this (such as 802.11n, which uses $40Mhz$-wide channels). Likewise, the channel selection and synchronisation protocols require reliable communication which, on an already severely affected channel, may not be possible. Therefore, frequency avoidance solutions should ideally be complemented by other approaches, either resilience, or detection.

Resilience solutions increase the likelihood of WSN data being received correctly in the presence of interference. Liang et al present a resilience approach based on observations of CTI between WiFi networks and WSNs [LPLT10]; this approach embeds sufficient redundancy to allow data packets to be recovered after a collision. Similarly, post-collision solutions describe a method to recover partially corrupted packets [HWW10, JB07]. These solutions have been shown to improve link performance in interference conditions, however do not mitigate energy inefficiency - stemming from false wakeups in WSN MAC protocols - caused by interference.

Detection approaches facilitate greater sensitivity of wireless devices to heterogeneous interference, and may focus on either WSN-, or interferer-detection improvement. In the

case of 802.11 - less sensitive and more destructive to other networks - interferer-detection provides the greatest link performance for 802.15.4. However, the complexity of this approach incurs significant hardware cost. Likewise, implementing this solution in existing $2.4Ghz$ deployments, such as WiFi, is practically unfeasible. Conversely, WSN-detection approaches allow for more selective contention policies in the WSN MAC. While not able to mitigate collisions with higher-powered interferers - such as WiFi - packet loss due to transmit FIFO overflow can be reduced. WSN-detection is also able to minimise energy consumption of LPL MAC protocols in interference environments, by reducing false wakeups. These approaches typically do not require any special radio hardware and can be implemented on most WSN platforms.

Sha et al [SHL13] and Tang et al [TWMM13] have presented solutions that tune the CCA threshold to meet energy consumption and packet delivery rate goals. These approaches suffice in environments where signal strength alone is sufficient to differentiate WSN traffic from interference. In more dynamic environments however, this is not the case [ZCW+14]. Zheng et al have proposed ZiSense, which detects 802.15.4 transmissions based on temporal RSS features, irrespective of signal strength [ZCW+14]. ZiSense is shown to reduce false wakeups in CTI environments, however, the idle listening - hence minimum energy consumption - is significantly increased. ZiSense is also not able to mitigate link quality degradation.

This review of current CTI solutions suggests that an efficient, low-cost approach may be offered by WSN-based detection works. These require no additional hardware, no modifications to existing heterogeneous networks, and can be implemented alongside existing MAC protocols. However, current state-of-the-art implementations incur significantly increased idle listening over standard MAC implementations. This reduces energy efficiency, and the battery life of nodes in the network. It would be beneficial to implement a WSN-based detection solution that does suffer this drawback, while still able to improve link quality and energy efficiency.

3.5 Chapter Summary

This chapter has examined existing work related to wireless coexistence, particularly with regard to 802.15.4 and WSNs. In section 3.2, previous empirical studies of CTI were reviewed. These measured the effects of interference on 802.15.4 devices and WSNs, and included coarse measurements in large deployments and fine observations in controlled experiments. These works have shown that CTI is detrimental to 802.15.4 links, resulting in increased packet loss and latency. For WSN MAC protocols this was shown to be exacerbated - reducing energy efficiency and network lifetime. These studies have highlighted the need to further study how to mitigate CTI.

In section 3.3, current methods to estimate energy consumption were reviewed. These are based on either simulation or theoretical models of energy consumption. The latter encompass models of the MAC and routing protocols, network load, hardware components, and physical deployment, among others. For example, using these models, it is possible to estimate energy consumption and network lifetime for a given deployment in ideal conditions. None of these, however, are able to account for the effects of CTI on WSN MAC protocols. Given that this effect is known to be non-negligible, it is therefore not possible using current techniques to estimate energy consumption and lifetime of a WSN in an interference environment. This has underscored the first problem presented in section 1.1: accurate estimation of WSN energy consumption and node lifetime is not possible.

In section 3.4, existing solutions that aim to mitigate the effects of CTI in WSNs were reviewed. A taxonomy of these works was presented, based on the underlying mitigation approach, including: avoidance, resilience, and detection. Avoidance solutions communicate on an unaffected channel, mitigating the effects of CTI entirely. Current implementations however assume a minimum link quality to coordinate channel selection, which may not be possible. Detection solutions offer the ability to mitigate packet loss, and improve energy efficiency in interference environments. Currently available detection solutions however either increase idle listening - and therefore energy consumption, or require impractical hardware changes. Consequently, there is currently no detection

solution available that remains energy efficient, and is possible on commodity hardware. This underscores the second problem presented in section 1.1.

Chapter 4

Estimating Node Lifetime

Predictable energy consumption is an invaluable component of WSN design. As discussed in Chapter 2, the receiving function of WSN MAC protocols is based on the 802.15.4 CCA interface, which detects incoming packets without costly-idle listening. However, the most common CCA implementation, energy-detection, is susceptible to other interference; energy consumption is therefore dependent on the environmental interference.

Consequently, WSN designers would benefit from a tool to estimate energy consumption of a WSN, based on interference measurements in an environment. This may be used to assess the feasibility of a deployment; fine tune MAC protocol parameters to meet network lifetime requirements; and compare MAC protocol performance.

To address problem P.2, raised in section 1.1, methods of measuring interference are first discussed in section 4.2. Then in Sections 4.3 and 4.4, two methods of predicting energy consumption are described, and shown by example for a well known MAC protocol. Prediction accuracy is then evaluated in Section 4.5. Finally, the chapter is summarised in Section 4.6.

4.1 Estimating Energy Consumption in WSN

When designing or installing a WSN, node energy consumption is a key design parameter. On battery powered devices, node energy consumption dictates network lifetime, and so WSN designers can tailor energy consumption to meet lifetime goals. A WSN deployment

to monitor volcano emissions, for example, may require that readings are recorded for at least three months. Alternatively, on WSN hardware powered by energy harvesting devices, node energy use must fall within the set budget for the network to operate. For example, a sensor node powered by a solar cell must not exceed this energy budget, in order for the node to function.

On most WSN hardware, the radio is the greatest source of energy consumption, even when not transmitting or receiving. Thus, node energy consumption and lifetime is determined more by the radio than any other component. However, as reviewed in Chapter 3, radio energy use has been shown in previous works to be influenced by environmental interference.

This is the case in asynchronous low power MAC protocols that infer from channel energy if a packet is being sent. In protocols such as ContikiMAC, and Tiny OS LPL, Clear Channel Assessments (CCA) are used to sample the channel energy and determine if a packet is being transmitted. This achieves very low energy usage, as the radio only briefly needs to be listening to the channel to determine if a further wakeup is needed. Most transceivers provide CCA based on energy detection, whereby the channel energy is sampled, and compared to a predefined threshold. This can detect 802.15.4 activity, and also any other activity in the 2.4Ghz domain, including other interference devices. In the context of a WSN MAC protocol wakeup sequence, this is liable to *false-wakeups*: where a WSN node infers channel activity from other interference. This increases idle listening and reduces energy efficiency and node lifetime.

Therefore, accurate predictions of WSN energy consumption, for these protocols, cannot be made without factoring the interference in the deployment. Tools and methods to achieve this can be used to:

1. **Tune MAC parameters** Wakeup frequency, CCA check behaviour, CCA threshold, and energy saving optimisations can be tuned to meet node energy use requirements for a given environment.

2. **Assess deployment feasibility** Given known hardware, network, application, and lifetime requirements, the feasibility of a WSN deployment operating in an

environment may be determined in advance.

3. **Comparing MAC protocols** Different MAC protocols may be compared on an energy efficiency basis, to select the most suitable for a deployment.

4.2 Measuring Interference

In order to estimate the energy consumption of a sensor node in a given environment, a quantification of the interference must be provided. This will be used as input to the energy consumption estimation method, and must accurately reflect the interference in an environment. Methods of sampling and analysing interference within 802.15.4 channels have received attention in WSN literature. This has been used for applications including identifying interference sources within the environment; optimising packet delivery in interference environments; and as input to channel selection algorithms. Methods of measuring interference in an environment, based on current WSN hardware, are discussed in this section.

A prominent approach has been to record the RSSI register value repeatedly into an in-memory buffer, providing a time domain trace of channel energy. This provides a detailed representation of any interference signals present, from which a wealth of information can be drawn. For each signal recorded, modulation characteristics, signal strength, and timing features may be calculated from such a trace. A high sampling frequency here is desirable to capture an RSSI trace representative of channel activity.

This method, however, suffers a number of drawbacks. Firstly, the memory capacity on typical WSN hardware is less than $10KB$, which limits the maximum sample duration. While compression techniques can mitigate this restriction, long uninterrupted sample durations are not possible. Likewise in applications which require this feature alongside other WSN functions, such a substantial memory allocation is not possible. Although this may not be an issue in a WSN tool set which has only this specific requirement. This approach is useful in applications which are required to analyse each detected signal or packet, such as interference source classification.

Statistical methods are able to compress the recorded RSSI before it is saved into memory. Some signal features are inevitably lost in this approach, although this is acceptable in applications which do not analyse each detected signal or packet - as in the case presented here. The distribution of idle/busy period frequencies in an RSSI trace has been adopted in previous literature. In this case, the idle and busy period durations of interference signals are divided into set defined ranges. On detection of each idle/busy period, the corresponding counter is incremented. In this approach, long sampling durations can be afforded at the expense of additional processing in each sampling loop iteration. The granularity of this approach can be tailored by altering the bin size. Packet delivery estimations, which are based on the probability of the channel remaining clear for a given packet length, have adopted this approach in previous work.

False wakeups in WSN MAC protocols stem from detection of interference in CCA checks in the wakeup sequence. Consequently, estimations of radio use stemming from false wakeups are required only to consider the interaction between interference and these CCA checks. In current state-of-the-art WSN MAC protocols, CCA checks within the CCA sequence are sufficiently separated, as to be considered temporally independent. For example, in the ContikiMAC protocol, CCA checks are spaced $500ms$ apart, which is far greater than the channel use of other interference sources. Therefore, in this chapter, the channel busy probability, denoted P_C, is used to measure interference. This is the probability of the channel being occupied by other interference at any instant, and is calculated as in equation 4.1.

$$P_C = \frac{\sum_{n=1}^{N} C_n}{N} \tag{4.1}$$

Where C_n is the state of the channel as busy (1) or clear (0) as measured on the nth sampling iteration, and N is the total number of samples recorded in a measurement. A quieter interference environment will be close to zero, while a busy channel will be closer to one. Depending on the interference source, requirements on the PHY and MAC layer within the standard may restrict this. For example, under 802.11g interference, P_C cannot exceed 55%, even under multiple stations transmitting at once. This is due to

the design of the 802.11g MAC protocol.

The memory and processing overhead of this approach is the smallest of these approaches, and can easily be incorporated alongside other WSN applications. As with measuring idle/busy distribution, only the channel state is required. This can be measured via CCA, which is much faster than RSSI and therefore supports higher sampling frequencies. This approach is sufficient for modelling WSN MAC protocol behaviour, based on the CCA wakeup sequence. However, it is insufficient for modelling more complex MAC protocol features, such as packet delivery. The chosen sampling frequency and duration affects the accuracy of this measurement. Increased sampling frequency will capture more interference signals and details, while increased measurement duration will better reflect the interference within an environment.

P_C is not expected to be static over time. Rather, it should increase during busier periods as devices generate more 2.4Ghz interference (for example, during working hours in an office), and decrease during quieter hours. However, the maximum recorded value of P_C can be used to calculate the worst case estimate for the energy use. Likewise, if the sample window for P_C is large enough, the average value of P_C can be used to estimate the average-case lifetime of the sensor node.

4.3 Closed Form Solution

In this section, a method to derive a closed form solution - $D(P_C)$, which is used to estimate idle listening time for a given MAC protocol - is described. This method was chosen firstly because the derived solution has minimal computing requirements to estimate idle listening time - and therefore may benefit embedded software environments. Secondly, this method may provide insights into interaction between the MAC protocol and interference. This may, consequently, aid in designing mitigation strategies.

This method excludes other factors which may affect energy consumption, such as hardware characteristics, internal WSN communication, and other node processing. Consequently, this method can be used to measure the impact on energy consumption based on changes to P_C. However, this approach may reduce the accuracy of the

estimation in real deployments.

A closed form solution, $D(P_C)$, takes as input the channel busy probability, P_C. D expressly ignores traffic within the WSN, assumes that the largest expenditure in energy consumption is idle listening. The idle listening time is a product of the expected radio on time, per wakeup sequence, E, and the channel check frequency f, as in equation 4.2.

$$D(P_C) = E(P_C) \cdot f \tag{4.2}$$

f is the rate that nodes check for incoming traffic, and is configured before the network is deployed; higher f allows for greater network throughput, but increases idle listening. E must be calculated for the MAC protocol, accounting for the wakeup procedure. As an example, a derivation of $E(P_C)$ for the ContikiMAC MAC protocol is now described. The operation of ContikiMAC is similar to other MAC protocols that use CCA to detect incoming traffic, to which this approach should be applicable.

The ContikiMAC wakeup sequence - as per the current implementation - is shown as a flow chart in figure 4.1. It consists of two components: the listen and receive phases. Firstly, in the listen phase, two CCA checks sample the channel to detect incoming traffic. The duration between the CCA checks is T_w, by default $500ms$, and is sufficient to ensure that at least one will coincide with any ongoing packet transmission. If either indicate a busy channel, the node enters the receive phase. Otherwise, the wakeup sequence terminates.

The receive phase is where the packet is received, after which an acknowledgement packet is sent (if required) and the radio is powered down. While waiting for a packet, the CCA and incoming data flag are polled continuously every $620\mu s$ to check the channel state. If no packet is received, the receive phase times out after N_{max} iterations, and the radio returns to sleep. The default setting of N_{max} is ten. To mitigate the cost of idle listening caused by false wakeups, *fast-sleep*, is described as an optimisation in ContikiMAC. Here, if the channel is found clear for N_{sil} uninterrupted CCA checks, a false-wakeup is assumed, and the radio is powered down. The default setting of N_{sil} is five.

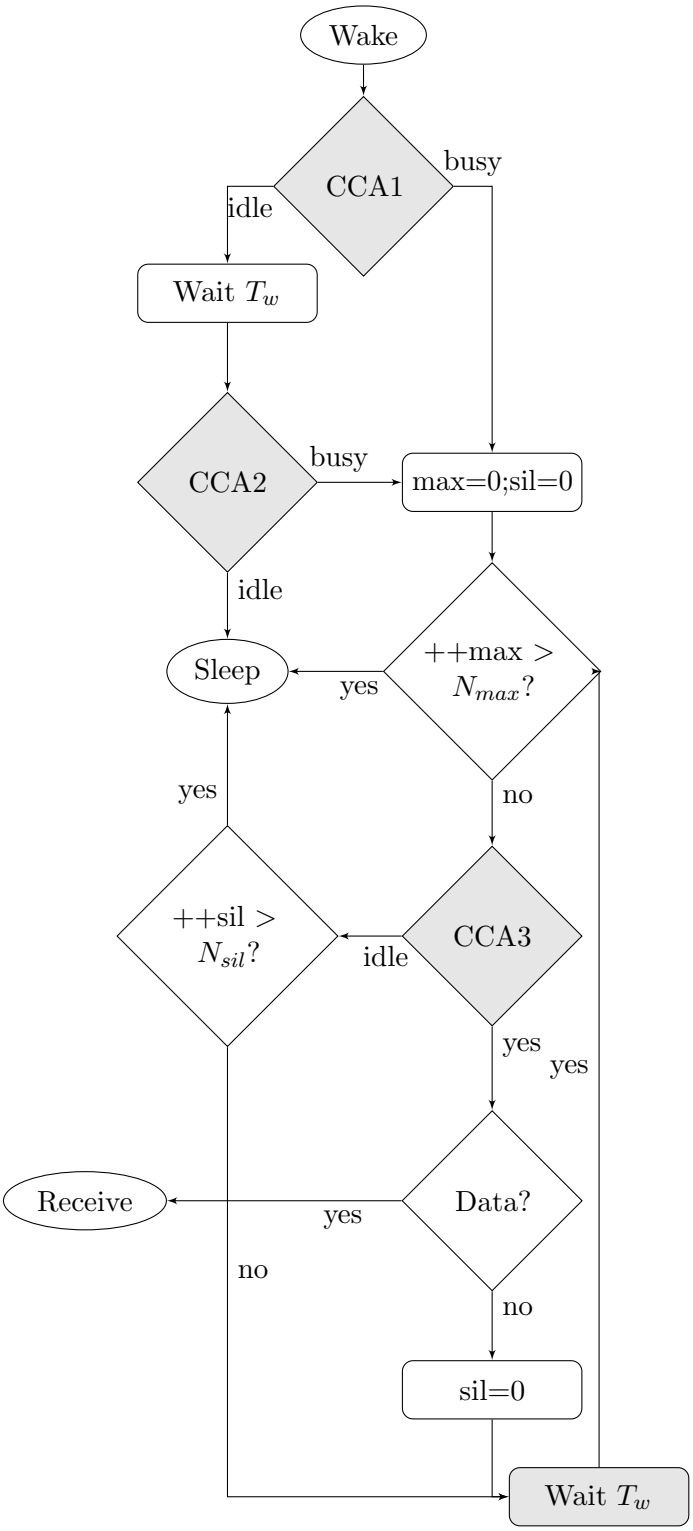


Figure 4.1: State diagram for ContikiMAC channel check sequence. Elements contributing to the idle duty cycle are shaded gray.

From this, the function $E(p)$ (see equation 4.3) is the sum of three terms, which are derived from the three paths the wakeup sequence can take in response to each CCA outcome. Each term is the product of the probability of its occurrence, and the resulting radio-on time in each case.

$$E(p) = E_{ii} + E_b + E_{ib} \tag{4.3}$$

These terms are collectively exhaustive, and encapsulate all code paths where no packet is received. E_{ii} represents the case where both CCA in the listen phase find a clear channel. No packet is detected, and there is no following false wakeup sequence. This is the optimal case, which achieves the lowest duty cycle. The probability of this branch executing within the state machine is $P_{ii} = (1-p)^2$. The radio-on time in this case is the time required to execute two CCA checks (CCA1 and CCA2), with duration T_1 and T_2. E_{ii} is thus given by:

$$E_{ii}(p) = (1-p)^2 \cdot ((T_1 + T_2)) \tag{4.4}$$

E_b represents the case where ContikiMAC's first CCA returns busy and ContikiMAC enters the receive phase of the wakeup sequence, wherein the channel is periodically checked via a CCA (CCA3 with duration T_3). In this procedure the node evaluates if a detected channel activity is part of an incoming transmission. A maximum number of $N_{max} + 1$ CCA's are carried out. The procedure may terminate before $N_{max} + 1$ CCA's are carried out if $N_{sil} + 1$ consecutive clear CCA's are encountered. Between each CCA a delay of T_w is included, which contributes to the radio on time as ContikiMAC keeps the radio active during the entire procedure. The very first CCA in this procedure returns always busy as there is no time delay between this CCA and the busy CCA leading into this procedure.

The default ContikiMAC configuration sets $N_{max} = 10$, and $N_{sil} = 5$. The time delays in the protocol which contribute to the radio-on time are, firstly, T_w, which is the time between channel checks in the receive phase. By default, this is set to $500ms$.

Secondly, T_1 and T_2 are the durations of the initial CCA checks in the listen phase. These were both measured as $294\mu s$. T_3 is the duration of the CCA check within the receive phase, measured as $122\mu s$.

For these settings, seven possibilities exist for the procedure to terminate before the maximum number of $N_{max} + 1 = 11$ CCA checks are carried out. For example, after the first CCA in the procedure – which always returns busy – we could encounter a sequence of 6 idle CCA which leads to a termination of the procedure after 7 CCA checks. The probability of this path is given by $p \cdot (1-p)^6$.

In equation 4.5, E_b is therefore given by the sum of three, collectively exhaustive, terms which represent all paths through the state machine.

$$E_b(p) = p \cdot \big(T_1 + E_1(p) + E_2(p) + E_3(p)\big) \tag{4.5}$$

In equation 4.6, E_1 represents the case where the first N_{sil} CCA checks are clear, and the $fast\ sleep$ mechanism powers down the radio.

$$E_1(p) = (1-p)^{N_{sil}} \cdot N_{sil} \cdot (T_3 + T_w) \tag{4.6}$$

In equation 4.7, E_2 represents the case where N_{sil} clear CCA checks are preceeded by up to $N_{max} - N_{sil} - 2$ other CCA checks. Here, the $fast\ sleep$ mechanism powers down the radio.

$$\begin{aligned} E_2(p) \quad = \quad & (1-p)^{N_{sil}} \cdot \Big(\sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \\ & \big(p^n \cdot (1-p)^{m-n}\big) \cdot (N_{sil} + m) \cdot (T_3 + T_w)\Big) \end{aligned} \tag{4.7}$$

Finally, equation 4.7 gives E_3, which is the expected duration if N_sil clear CCA checks are not found, and the $fast\ sleep$ mechanism is not enacted.

$$\begin{aligned} E_3(p) \quad = \quad & \Big(1 - \big((1-p)^{N_{sil}} + \sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \\ & p^n \cdot (1-p)^{m-n}\big)\Big) \cdot (N_{max} - 1) \cdot (T_3 + T_w) \end{aligned} \tag{4.8}$$

Combining these terms, as in equation 4.5, E_b can be given by:

$$\begin{aligned} E_b(p) &= p \cdot (1-p)^{N_{sil}} \cdot \Big((N_{sil} \cdot (T_3 + T_w) + T_1) \\ &+ \sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \big[p^n \cdot (1-p)^{(m-n)} \\ &\cdot \big((N_{sil}+m) \cdot (T_3+T_w) + T_1 \big) \big] \Big) \\ &+ p \cdot \Big(1 - \big((1-p)^{N_{sil}} \\ &+ \sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \big[p^n \cdot (1-p)^{(m-n)} \big] \big) \Big) \\ &\cdot \big((N_{max}-1) \cdot (T_3+T_w) + T_1 \big) \end{aligned} \tag{4.9}$$

E_{ib} represents the case where ContikiMAC's first CCA (CCA1) returns clear but the second CCA (CCA2) returns busy which then leads to the execution of the same procedure as described for E_b. The difference here is the resulting duration of the radio on time, as two CCA are executed before entering the receive phase. E_{ib} is therefore given in equation 4.10.

$$E_b(p) = (p-1) \cdot p \cdot \big(T_1 + T_2 + E_1(p) + E_2(p) + E_3(p) \big) \tag{4.10}$$

As above, the expansion of E_{ib}, is given by:

$$\begin{aligned} E_{ib}(p) &= p \cdot (1-p)^{N_{sil}+1} \cdot \Big((N_{sil} \cdot (T_3 + T_w) + \\ &T_1 + T_2) \\ &+ \sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \big[p^n \cdot (1-p)^{(m-n)} \\ &\cdot \big((N_{sil}+m) \cdot (T_3+T_w) + T_1 + T_2 \big) \big] \Big) \\ &+ p \cdot \Big(1 - \big((1-p)^{(N_{sil}+1)} \\ &+ \sum_{m=1}^{(N_{max}-N_{sil}-2)} \sum_{n=1}^{m} \big[p^n \cdot (1-p)^{(m-n)} \big] \big) \Big) \\ &\cdot \big((N_{max}-1) \cdot (T_3+T_w) + T_1 + T_2 \big) \end{aligned} \tag{4.11}$$

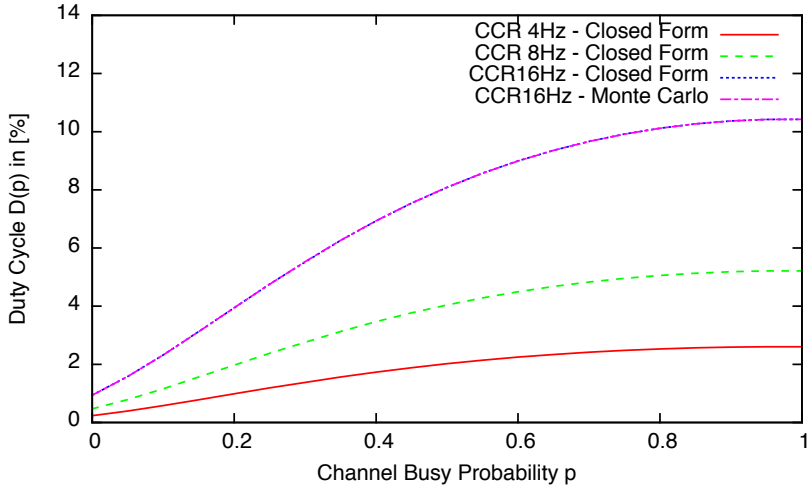


Figure 4.2: Closed form estimation of ContikiMAC duty cycle under interference conditions

The predicted duty cycle function, $D(P_C)$, of ContikiMAC is plotted in figure 4.2, for channel check rates $4Hz$, $8Hz$, and $16Hz$. The graph shows, firstly, that $D(P_C)$ increases with channel busy probability, p. This is due to false wakeups, which are caused by CCA checks in the wakeup sequence detecting interference. Consequently, this model predicts that node energy consumption is a function of environmental interference. This correctly meets the expected behaviour of this WSN MAC protocol under interference, and therefore the validity of the model is supported.

Figure 4.2 secondly shows that higher channel check frequencies, f, increase energy consumption and exacerbate the impact of p on node energy consumption. This leads to the finding that, in order to improve energy efficiency and network lifetime, WSNs deployed in interference environments should minimise f, as part of the design phase.

As p increases toward one, the duty cycle begins to plateau. This is due to the asymptotic nature of the model. For example, for $f = 8Hz$, which is a common setting in WSN, the optimal duty cycle is 0.5% under ideal conditions without interference ($P_C = 0$). Under worst-case conditions ($P_C = 1$), the duty cycle increases to approximately 4.5% - 11 times worse. Obviously, in such circumstances WSN communication is not possible, but the model shows that interference levels must be considered when estimating network lifetime.

4.4 Monte Carlo Solver

For a given value of P_C, the closed form solution $D(P_C)$ offers precise estimations of the duty cycle of affected nodes. However, the difficulty of deriving $D(P_C)$ for a MAC protocol is dependant on the complexity of the wakeup sequence. The implementation of ContikiMAC is quite simple, however, deriving $D(P_C)$ is difficult, due in part to the fast sleep mechanism. Even small modifications to ContikiMAC, such as changing the number of CCA checks used in the periodic channel check, or allowing N_{mac} to vary in response to P_C, for example, would require significant changes to $E(P_C)$. Other MAC protocols, which use more complex mechanisms, introduce more paths and assigned probabilities through the state diagram, which $D(P_C)$ would need to account for.

A more flexible solution, which can be readily adapted to new changes and MAC protocols, is desirable. To achieve this, a Monte Carlo solver was used. Here, the MAC protocol wakeup sequence is executed in a simulator, from which the radio-on time is measured. To determine the expected radio-on time, $E(p)$, the simulator is executed N times. For each run, the calculated radio-on time, $E_n(p)$ is recorded. The expected radio-on time is then the average, calculated as:

$$D(p) = \frac{f}{N} \sum_{n=1}^{N} E_n(p) \tag{4.12}$$

Following the previous section, the ContikiMAC wakeup sequence was implemented in the simulator. The simulator, which was written in the scripting language Lua, requires as input the channel busy probability, p. This is incorporated into the simulation as a function call for the CCA, which returns true/false based on the probability p. The solver takes as input also the channel check frequency, f, and outputs the expected duty-cycle for the wakeup sequence, $D(p)$.

Algorithm 1 shows the algorithm for ContikiMAC in the simulator. This implementation is based on the protocol state machine, shown in figure 4.1.

This approach is much closer to the actual implementation of ContikiMAC, and so is trivial to implement. The ContikiMAC protocol builds atop the Contiki OS, which

Input: busy probability: p
Result: radio on time: E

E = 0; first = TRUE; silence = 0; count = 0;

E += T_1;
if *cca_clear(p) == TRUE* **then**
  E += T_2;
  if *cca_clear(p) == TRUE* **then**
    return E;
  end
end

while *TRUE* **do**
  if *first == TRUE OR cca_clear(p) == FALSE* **then**
    silence = 0;
  else
    silence++;
  end
  first = FALSE;
  count++;

  if *(silence* $> N_{sil}$ *) OR (count* $> N_{max}$ *)* **then**
    return E;
  end

  E += $T_3 + T_w$;
end

Algorithm 1: The Monte Carlo solver simulation of the ContikiMAC state machine

provides a thread-based abstraction for sleeps and delays. In OSs that lack this feature, or are event-based, a different approach may be appropriate.

The output of the Monte Carlo solver was measured for $f = 16Hz$, and included in the figure 4.2. Comparison of the result with the result provided by the closed form solution shows little deviation. The largest deviation is 0.25%, with a busy probability of $p = 0.3\%$. Hence, the Monte Carlo solution was concluded to offer comparable precision of energy consumption estimation as the closed form solution, while still providing flexibility to handle different MAC protocol parameters.

4.5 Evaluation

In this section, the evaluation of the Monte Carlo solver and closed form solution methods is described. Three experiments were carried out. The first took place under controlled interference, and evaluated the accuracy of energy consumption estimation under various WiFi interference rates. The second experiment took place in two uncontrolled interference environments, and again measured the prediction accuracy under realistic interference. The final experiment measured also the effect of WSN traffic on the accuracy of these estimations. The objective of these experiments was to measure accuracy of the predicted duty cycle, as compared to the measured duty cycle in the WSN.

From sections 4.3 and 4.4, the Monte Carlo solution was shown to be the simplest to measure in practice. Therefore, the Monte Carlo solution is used in this evaluation. In section 4.4, it was shown that, for the same input values of p, the estimated duty cycle of the Monte Carlo solver and the closed form solution are similar. Therefore, the following evaluation of the Monte Carlo solution is also applicable to the closed form solution.

4.5.1 Controlled Interference

The objective of the first experiment was to evaluate the accuracy of the Monte Carlo solver under controlled interference conditions. In this experiment, two Tmote Sky sensor nodes and an 802.11g network, which consisted of an access point and station, were deployed in an unused office.

The methods to estimate idle listening presented earlier are based only on the CCA, and not on the specific Received Signal Strength (RSS) of signals. Therefore, in this experiment, the transmission power of the 802.11g interferers was set to the maximum, $20dBm$. The distance between the WSN nodes and the 802.11g network was $6m$. This was to ensure that all 802.11g packets generated exceeded the CCA threshold.

The ContikiMAC protocol, described previously, ran on one of the nodes using the default parameters, including channel check frequency, $f = 8Hz$. As the node was running, the cumulative radio-on time was calculated for each experiment run. No packets were sent from the node, hence the radio-on time is caused only by idle listening.

Busy Probability p [%]	Data Rate [kbit/s]
1.74	0
3.91	500
7.55	2000
13.10	4000
23.81	8000

Table 4.1: The mapping of WiFi data rates and measured channel busy probability p.

The second node measured the channel busy probability, p, throughout each experiment - without any MAC protocol or duty cycling. This was used to estimate idle listening based on the Monte Carlo solver method. In both cases, the results were communicated to the host computer via serial connection.

The 802.11 network generated controlled interference using the iperf tool [TQD+05] on an 802.11 channel adjacent to the 802.15.4 channel used by the WSN. The corresponding value for p, for each interference level as measured on the Tmote Sky node, is shown in table 4.1. 802.11g traffic rates were used corresponding to a channel busy probability range from 1.74% to 23.81%. It was not possible to obtain an entirely interference-free environment ($p = 0$), due to background interference. This is inconsequential though, as only p is used as the independent variable, not the specific traffic rate. Higher interference levels were not used as WSN deployments are unlikely in such environments. Due to this experiment having no initialisation phase or stabilisation delay, the results were recorded in short intervals of five minutes.

The measured value for p is plotted against the output of the Monte carlo simulation and the actual measured duty cycle in figure 4.3. The results show that the Monte Carlo solver closely estimates the recorded duty cycle. The largest deviation from the recorded measurement is 7.4%, for $p = 7.55\%$. Therefore, this approach has been shown to provide high prediction accuracy of estimating the radio duty cycle of ContikiMAC, under known interference conditions.

The relationship between the recorded channel busy probability, p, and the duty cycle appears linear. This experiment further supports the conclusion to sections 4.3 and 4.4: that energy efficiency is determined by interference conditions. In this experiment p was

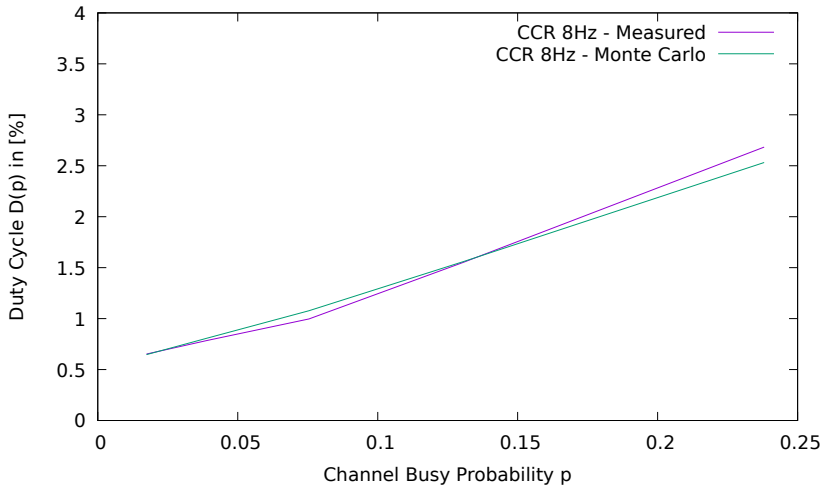


Figure 4.3: ContikiMAC duty cycle under controlled interference settings.

known and could be compared against the actual recorded duty cycle. In a practical deployment, p may only be measured once, to estimate the duty cycle.

4.5.2 Uncontrolled Interference

The previous experiment evaluated the accuracy of the energy estimation under controlled interference, for a particular interference type. In real deployments, WSNs may be subject to a larger variety of interference sources, including Bluetooth and MWO. This may affect the accuracy of the estimation technique. Therefore, the objective of this experiment is to measure the prediction accuracy of the Monte Carlo solver under uncontrolled interference, which is more representative of realistic interference conditions.

In two deployments two sensor nodes were deployed as previously. One Tmote Sky node recorded the interference in the environment, measured as the channel busy probability, p. This was used to estimate the duty cycle of ContikiMAC via the Monte Carlo solver. The second Tmote Sky node implemented the ContikiMAC protocol, and periodically recorded the duty cycle of the radio. No WSN traffic was generated by either node, and therefore all radio activity stemmed from the wakeup sequence in ContikiMAC.

In order to capture a range of interference representative of urban environments, two locations were chosen: a meeting room and a shared office. These were subject to various interference sources, including WiFi and Bluetooth, throughout the experiment due to

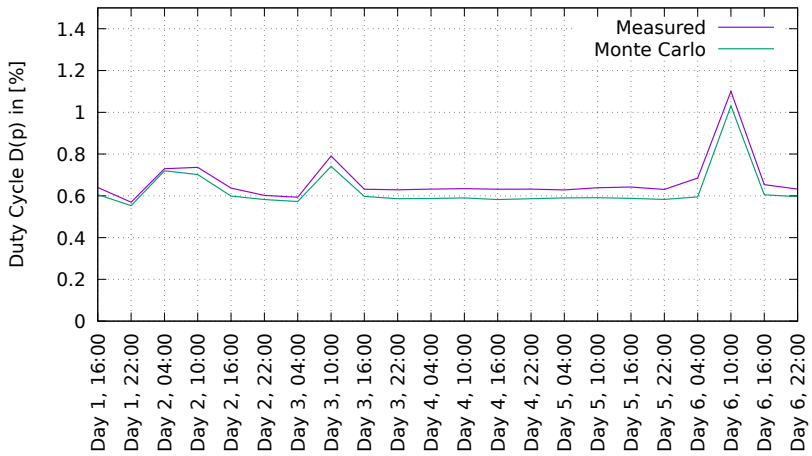


Figure 4.4: ContikiMAC's duty cycle estimated and measured over time ($f = 8$). This experiment was carried out in a meeting room.

normal activity.

The host computer communicated with both nodes via serial connection, and calculated the six-hour average of estimated and recorded duty cycle. In both nodes, the CCA threshold was set to the default value. The results from both deployments are plotted in figures 4.4 and 4.5.

These graphs show firstly that the duty cycle of ContikiMAC varies throughout the experiment. This is due to normal activity, such as staff arriving, attending meetings, and working. The largest periods of activity happen on days two, six, and eight, during working hours. Interference is not consistent throughout the experiment, and is much lower on other days. Consequently on days four and five, there is no change to the duty cycle.

These graphs show that the predicted duty cycle, as calculated by the Monte Carlo solver, closely follows the observed duty cycle throughout the experiment. The average deviation of the predicted duty cycle from the measured duty cycle is 6.23% and 2.09% for the office and meeting room respectively. In the office deployment, the worst case deviation was 13.1%, on day six. In the meeting room deployment, the worse case was 12.94%, on the second day.

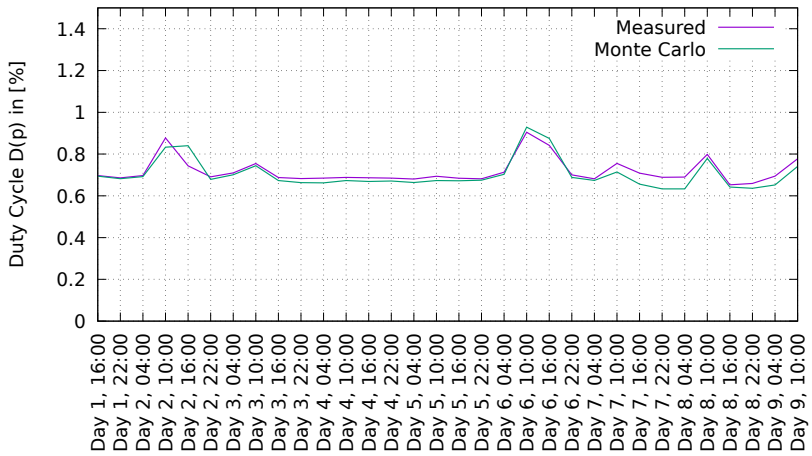


Figure 4.5: ContikiMAC's duty cycle estimated and measured over time ($f = 8$). This experiment was carried out in an office.

4.5.3 Background WSN Traffic

Experiments in Sections 4.5.1 and 4.5.2 have shown that the duty cycle of ContikiMAC can be accurately predicted based on the Monte Carlo solver. These experiments were tested under the premise that idle listening constitutes the only use of the radio, hence why previous experiments did not involve any WSN traffic. Additional packet transmissions will increase the radio-on time, which is not accounted for in the prediction. The objective of this experiment is to evaluate this hypothesis.

In this experiment, three Tmote Sky nodes were placed $6m$ apart in an unused office, alongside an 802.11g network at a distance of $6m$.

As previously, one sensor node recorded the channel busy probability, p, throughout the experiment. A second sensor node executed the ContikiMAC protocol, and recorded the duty cycle of the radio. This node also transmitted packets to the third node at a variable rate, using the ContikiMAC send function. In this experiment, a packet size of 120 bytes was chosen, since this is the largest packet size supported. Consequently, this provides an upper bound, to test this hypothesis. Only the packet rate of the link layer, and not higher network layers, was used as independent variable, and therefore retransmissions are not implemented. Similarly, a routing protocol was not used, as this was a single hop network. The channel check rate was set to $8Hz$, and the remaining

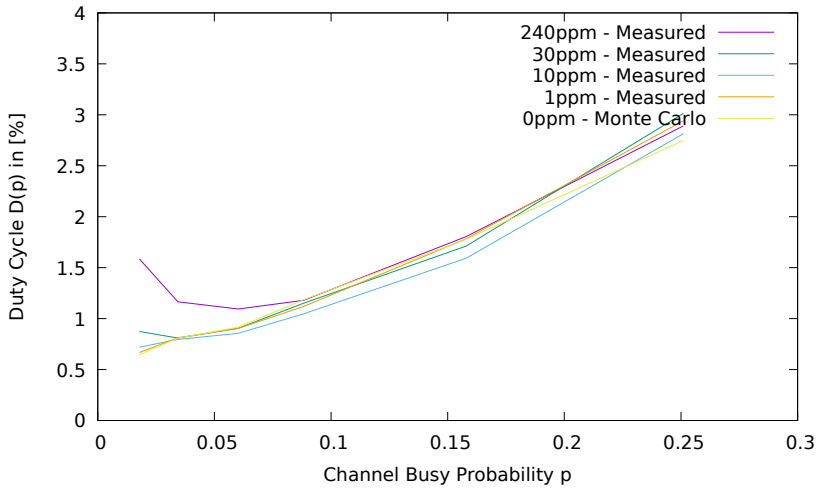


Figure 4.6: ContikiMAC duty cycle under controlled interference, for WSN traffic rates 1, 10, 30, and 240 packets per minute.

ContikiMAC parameters were set to default values.

The 802.11 network generated controlled interference using the iperf tool [TQD+05] on an 802.11 channel adjacent to the 802.15.4 channel used by the WSN. The same interference rates as in the first experiment were used here (see table 4.1). As previously, each interference rate was tested in five minute intervals. The predicted duty cycle, calculated from p, is plotted alongside the recorded duty cycle for packet transmit rates 1, 10, 30, and 240 packets per minute. The results are shown in figure 4.6.

The results show, firstly, that the trend of the estimated duty cycle closely predicts the recorded duty cycle for low packet transmission rates below $240ppm$. For packet transmission rates above this, the estimated duty cycle is below the recorded duty cycle. This is because of packet transmissions, which increase radio-on time and the duty cycle of the radio. This is not accounted for in the model.

Secondly, for all packet transmission rates, the estimated duty cycle is close to the recorded duty cycle for high interference levels, above $p = 0.1$. This is because, for high interference levels, the idle listening incurred by false wakeups dwarfs that incurred due to packet transmissions. Therefore, the affect on the duty cycle of packet transmissions is reduced.

In summary, the results in figure 4.6 show that the estimated duty cycle is accurate

for either low packet transmission rates, or high interference rates.

4.5.4 Discussion

These experiments have evaluated the prediction accuracy of the Monte Carlo solver, based on interference measurements. The results have shown that, firstly, in the presence of low WSN traffic rates or high interference levels, the trends of the estimated and the recorded duty cycle are similar. The results have shown that, with no WSN traffic, the estimated duty cycle is accurate to within 13.1%. Consequently, this supports the validity of the closed for solution and the Monte Carlo solver.

There is some deviation from the recorded and the estimated duty cycle, however. In figures 4.4 and 4.5, the predicted duty cycle underestimates the recorded duty cycle, which is not the case under controlled interference. In this case, WSN designers could underestimate the energy consumption of a network, and therefore overestimate network lifetime in real deployments.

These absolute differences could be the result of hardware or software components that affect the listen behaviour of the radio, but are not accounted for in the estimation. This could include, firstly, hardware differences in the CCA sensitivity between the two Tmote Sky nodes used in the experiment. This would be the case if, for example, one of the nodes has greater signal attenuation in the PCB antenna. In hindsight, it would have been beneficial to include a larger number of WSN nodes to reduce this effect. Secondly, this could be caused by interference sources that were not included in the controlled interference experiment, which only evaluated WiFi interference.

Differences between the estimated and the recorded duty could otherwise stem from timing inconsistencies in the implementation of ContikiMAC, which may not accurately follow the model shown in figure 4.1. This is due to the timing accuracy of the abstractions provided by Contiki OS, such as delays and interrupt handlers.

4.6 Chapter Summary

In chapter 3, it was shown that previous energy estimation techniques do not consider environmental interference, making it difficult to estimate energy consumption or node lifetime in such deployments. Accurate energy consumption estimation is important to validate the feasibility of a deployment, inform hardware selection, and fine tune protocol parameters.

In this chapter, a method to estimate WSN radio use and energy consumption in interference environments was sought after, in order to address the first problem, P.1, raised in section 1.1.

Firstly, methods to measure interference in a deployment were considered, ultimately choosing channel busy probability, P_C. Following this, a method to derive a closed form model to estimate the radio-on time of a MAC protocol was described, taking P_C as input. This is based on the design of the wakeup procedure for the MAC protocol. Based on this, energy consumption of the MAC protocol, and of the node, can be estimated. A Monte-Carlo approach was then presented, based on a MAC protocol simulator, to demonstrate an easier method to model radio use under CTI. In both cases, ContikiMAC - a common MAC protocol in WSN - was used as example.

The evaluation in Section 4.5 showed that the latter is able to achieve high prediction accuracy under controlled and uncontrolled interference, and under typical WSN traffic conditions.

Chapter 5

Improving WSN Channel Sensing

In this section, a method to mitigate the effects of CTI, as discussed in section 1.1, is sought after. The review of current state-of-the-art solutions to CTI in section 3.4 found that WSN-based detection approaches are able to effectively mitigate the effects of CTI, without impractical hardware modifications or protocol adaptions. Current solutions, however, were shown to greatly increase the idle listening cost, impairing energy efficiency.

Consequently, an extension to the standard 802.15.4 CCA is described in this chapter, in order to address problem P.2 raised in section 1.1, This can indicate the origin of an interference source, as well as the channel state. This mechanism, termed Differentiating CCA (DCCA), allows nodes to execute a more systematic response to interference, to mitigate energy inefficiency and packet loss caused by interference. Two implementations that are feasible on typical WSN hardware are discussed in Sections 5.2 and 5.3 respectively: Time-DCCA (T-DCCA), which is based on a similar approach in literature, and Power-DCCA (P-DCCA), presented in this thesis. The accuracy and energy consumption of these approaches is compared in Sections 5.4 and 5.5. P-DCCA is then implemented in the ContikiMAC protocol, and evaluated in a typical WSN context in Section 5.7. Finally, this chapter is concluded in Section 5.8.

5.1 Differentiating CCA

Contention-based low power WSN MAC protocols, including ContikiMAC and TinyOS LPL, were shown in Section 2.2 to use CCA for both sending and receiving data. On most 802.15.4 radio hardware, CCA is provided as ED, whereby the channel energy is measured, and compared to a preset threshold to infer the channel state. Since this cannot discern the origin of a signal, MAC protocols must implement a broad policy for all interference types; this is not optimal in terms of interference coexistence.

MAC protocols would benefit from a more informative indication of channel state, to include the origin of a detected signal. This would afford MAC protocols a more systematic response to collisions and interference. The receiver wakeup sequence could only be initiated after detecting WSN traffic, ignoring all other interference. This would reduce false wakeups, and improve energy efficiency. Likewise, senders could handle collisions differently depending on the nature of the channel contention.

Differentiating Clear Channel Assessment (DCCA) is proposed as a conceptual extension to the standard 802.15.4 CCA mechanism, capable of returning three channel states: 1) Collision with another device in the same network, 2) Collision with another device of a different network, 3) Clear Channel. In the following section, implementation options of DCCA on commodity 802.15.4 hardware are discussed, implemented, and evaluated.

5.1.1 DCCA Implementations

Three methods of realising DCCA on commodity WSN hardware were considered, and are described below. Each has tradeoffs in terms of the underlying hardware requirements, processing overhead, accuracy, and energy cost.

1. **Modulation Detection-DCCA (MD-DCCA):**

   Using hardware components to implement DCCA would be most efficient, as software resources could be devoted solely to the WSN application. As discussed in Section 2.1.2, the 802.15.4 standard defines carrier sense as an optional CCA method,

based on modulation detection. In conjunction with the RSSI, which measures the channel energy, the requirements of DCCA can be fulfilled by distinguishing between a clear channel, an 802.15.4 transmission, and non-802.15.4 interference. This would be ideal for implementing DCCA with little processing overhead.

Using the AVR RF230 radio, which provides carrier sense CCA, the accuracy of this CCA mode was measured in a lab environment. Almost 100% true positive, and 100% true negative accuracy was recorded, affirming that MD-DCCA would be a viable option.

Unfortunately, in order for transceivers to provide MD-DCCA, more complex radio circuitry is required. This increases the manufacturing cost, and also increases the energy consumption of receiving and idle listening, therefore reducing energy efficiency. As such, most currently available 802.15.4 transceivers do not implement this method, and opt instead only for ED CCA. This is the case with the Texas Instruments family of 802.15.4 transceivers, including the popular CC2420. Consequently, other methods are required to provide similar behaviour through novel means in software.

2. **Time-DCCA (T-DCCA):** 802.15.4 transmissions have different modulation and timing characteristics compared to other interference signals, which can be used for identification. These characteristics can be captured as a trace of signal power over time by sampling the RSSI register at high frequency. From this, the origin of a detected signal can be inferred, meeting the requirements of DCCA.

   This method is achievable on most 802.15.4 radio hardware, requiring only an RSSI register that can be sampled at high frequency. As well, this method requires no change to the transmitting function, reliant on the characteristics of the 802.15.4 physical layer and the MAC protocol used by the network. However, long listen durations may be necessary to capture the temporal RSSI characteristics necessary to identify 802.15.4 packets.

   ZiSense [ZCW+14] is an existing approach in literature which adopts this approach

to identify 802.15.4 packets amongst other interference. ZiSense is used as a reference implementation of T-DCCA in this thesis.

3. **Power-DCCA (P-DCCA)**:

   Building on T-DCCA, P-DCCA also samples RSSI at high frequency to identify prior-known signal characteristics. However, in order to reduce the required sampling duration, P-DCCA does not rely on inherent 802.15.4 features, such as spectral profiles and packet durations. Instead, P-DCCA relies on an additional signal component that is modulated by the transmitter, and can be detected by analysing an RSSI trace. This is orthogonal to the data transmission, and is used only to differentiate signals from interference. Since these features would be intentionally observable on a smaller timescale, the detection time would be reduced compared to T-DCCA.

   One implementation of P-DCCA, which is described in Section 5.3, achieves this by varying the output power cyclically throughout packet transmission. P-DCCA receivers sample the RSSI register at high frequency, detecting this encoded characteristic within the trace. Signals which lack this feature are assumed to be other interference.

   Compared to T-DCCA, the detection accuracy of P-DCCA could be improved by rendering the transmit power encoding sufficiently distinguishable from other interference. This approach would have a shorter detection, and thus would have a lower idle listening cost compared to T-DCCA. However, reducing the transmission power decreases the SNR of the packet at receivers, which may reduce the probability of the packet being received correctly. The ability to sample the RSSI register at high frequency, and vary the output power during packet transmission, are the only two hardware requirements.

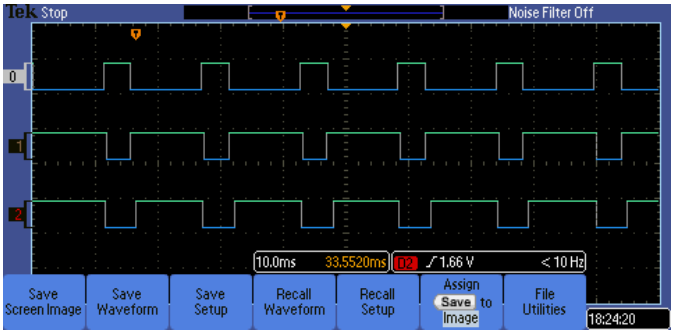


Figure 5.1: Screenshot from the oscilloscope recording the state packet transmission (probe 1), CCA Mode 2 (probe 2), CCA Mode 1 (probe 3).

5.1.2 Discussion

Being hardware based, MD-DCCA would likely provide the best detection accuracy, as well as the shortest detection time. As well, the implementation of MD-DCCA would be much simpler than the other methods described.

The CC2420 specification states that all three CCA options, described in Section 2.1.2, are implemented, including carrier sense. Under scrutiny, however, this is not the case. Instead, CCA Mode 2 (Carrier sense), indicates a clear channel *when not receiving valid 802.15.4 data* [Ins06]. In order for data to be received on the CC2420, or any 802.15.4 transceiver, it is necessary to detect the packet preamble and SFD. This is not useful to duty-cycled WSN protocols, since the packet preamble cannot be detected whilst the radio is powered off.

An experiment was run to confirm that this is the case. A Tmote Sky node was programmed to transmit packets, raising a General Purpose Input/Output (GPIO) pin when a packet is being transmitted. Two other nodes in close proximity were programmed to continuously sample the CCA pin of the CC2420 radio. These nodes were programmed to use CCA modes 1 (ED) and 2 (CS) respectively, and mirror the state of the pin on a separate GPIO pin. A logic high on the CC2420 by default indicates a clear channel, while a logic low indicates a busy channel. All three nodes were then connected to an oscilloscope, to monitor the state of the CCA simultaneously. A screenshot is shown in figure 5.1, where both CCA modes are shown to detect packets.

To confirm that the CC2420 is reliant upon detecting the preamble and correct SFD,

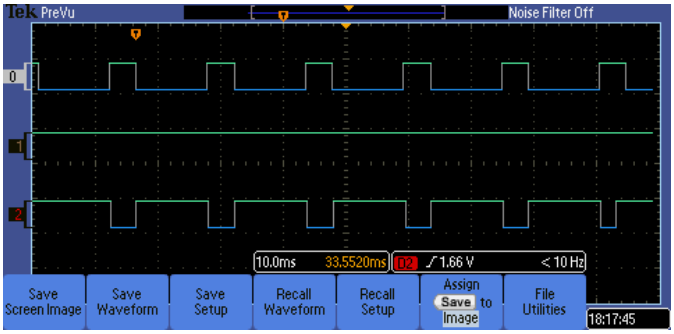


Figure 5.2: Screenshot from the oscilloscope recording the state of packet transmission, with different SFD value.

the two CCA-sampling nodes were programmed to change the value of the SFD, from the default 0x7A, to a different value, by modifying the SYNCWORD register. In this configuration, packets could no longer be received with the default SFD. The same experiment was repeated, and a screenshot from the oscilloscope is shown in figure 5.2. As shown, only CCA Mode 1 is able to detect packets. CCA Mode 2 is unable to detect anything, and indicates a clear channel. Consequently, as with other radio that lack true carrier sense, MD-DCCA is not possible on the CC2420 radio.

Conversely, T-DCCA and P-DCCA both search for known temporal characteristics in RSSI traces, which do not require atypical hardware. The use of explicit feature encoding in P-DCCA is assumed to make these signals more readily recognisable, potentially improving accuracy and reducing the required detection time. This is weighed against the cost of this power modulation: reducing the signal strength at receivers and increasing packet errors. Thus, the use of P-DCCA and T-DCCA is a tradeoff: shorter detection time of P-DCCA compared to T-DCCA, at the cost of higher error rates as consequence of the power modulation scheme.

The shorter detection time required by P-DCCA would be closer to the 802.15.4 standard CCA duration, and so would be more easily implemented on existing MAC protocols. Further, if the power variation sequence were unique enough, it is less likely that other sources of interference could be mistaken for P-DCCA transmissions. By contrast, the temporal signal characteristics relied upon by T-DCCA cannot be assumed unique when considering all possible interference sources.

The reduced link performance assumed in P-DCCA is mitigated when the operation of LPL-based MAC protocols is considered. These require that the signal strength of incoming packets exceed the preset CCA threshold, in order to be received. Since 802.15.4 receivers have notably good performance even near the noise floor, it is plausible that link performance is restricted sooner by this threshold, than by the performance of the physical layer.

The operation of P-DCCA and an implementation on the Tmote Sky is described in Section 5.3. The accuracy of this approach is then measured against T-DCCA, following an implementation based on ZiSense which is described in Section 5.2.

5.2 T-DCCA

For this work, the accuracy of the P-DCCA and T-DCCA mechanisms are evaluated in different interference conditions. The closest approach to T-DCCA in previous literature is ZiSense [ZCW+14], which is therefore used as a baseline here. ZiSense is designed to be incorporated into the wakeup sequence of low power WSN MAC protocols, to improve the detection of incoming packets. Although ZiSense was not directly designed as a DCCA mechanism, it can be employed in this capacity as representation of T-DCCA and is used in this thesis for comparison against P-DCCA.

ZiSense samples the RSSI register at high frequency into a buffer, generating a signal strength trace over time. This trace is analysed and segments are identified as contiguous subsets of RSSI samples when the signal differs from the noise floor. For each segment detected, a feature-set is constructed describing:

1. On-air time
2. Peak to Average Power Ratio (PAPR)
3. Inter Packet Spacing (IPS)
4. Under Noise Floor (UNF)

(1) and (3) describe temporal features of the signal, which are typically specific to the MAC protocol. The original authors of ZiSense used TinyOS LPL as the MAC protocol in their evaluation. (2) stems from the modulation used by the transmitter. (4) is an indicator if the RSSI drops significantly below the noise floor. This is used to detect MWO interference, wherein this phenomenon is caused by saturation of the intermediate amplified chain in the CC2420 transceiver.

ZiSense requires that the sampling frequency is sufficient to detect these features; the author's work is based on a $32Khz$ sample rate. As well, the sampling duration must at least match the IPS of the MAC protocol in use. The author's work is evaluated for the TinyOS LPL MAC protocol which requires a $2.9ms$ sampling period, 90 RSSI samples.

Having constructed a feature set for each segment detected within the sample, each is then classified as either 802.15.4 traffic, or other interference. In [ZCW+14], three algorithms are described: *ZiSense-1*, *ZiSense-2*, and *ZiSense-C4.5*.

- *ZiSense-1*: A classifier manually derived from strict interpretation of 802.15.4 PHY and TinyOS LPL.
- *ZiSense-2*: A more forgiving classifier manually derived to identify segments that have collided with other interference.
- *ZiSense-C4.5*: A classifier built using the C4.5 algorithm, which builds a decision tree classifier from a prior-known data set - in this case from segments which were identified manually.

T-DCCA detects the presence of incoming packets using ZiSense as described. If the algorithm indicates any 802.15.4 packets, a collision with another WSN device is indicated. Conversely, if the algorithm detects only non-802.15.4 signals, a collision with another interference source is indicated. An empty sample set, detecting no signals, indicates a clear channel.

Since no implementation of ZiSense is currently available, an offline approach was implemented based on the author's original description, in order to evaluate T-DCCA. An RSSI sample set was collected by the Tmote Sky hardware into a temporary 4000-

value buffer, at the same frequency as described in [ZCW+14]. Once full, sampling is paused whilst the contents of the buffer is transmitted to a host computer. The three ZiSense classification algorithms are then run on this data set, on each contiguous set of 90 samples. This approach avoids the complexity of implementing ZiSense on limited-resource hardware, while still being able to evaluate classification accuracy.

5.3 P-DCCA

P-DCCA consists of two components. Firstly, as packets are being transmitted, the radio is set to vary the output power in a set sequence. Secondly, this power modulation must be detected and correctly identified during a P-DCCA check.

802.15.4 radio transceivers provide an interface to configure the transmission power. On the TI CC2420, for example, a five-bit register setting permits values in the range 0-31, corresponding to the output power range from -25dBm to 0dBm. Experiments with two 802.15.4 radios, the TI CC2420 and also the AVR RF230, confirmed that transmit power can be altered during transmission.

The 802.15.4 standard requires radios to provide an RSSI interface for the purposes of channel selection; for example, to choose the most idle channel. As with the CCA, this is calculated by averaging the channel energy over time. On the TI CC2420, the average window, A_v, is fixed to 128μs, however, some other radios allow this to be customised. By sampling the RSSI register at high frequency, a trace of the RSSI in the time domain is generated. In a P-DCCA check, this trace is searched for features indicative of an 802.15.4 signal transmitted with P-DCCA power modulation.

The P-DCCA transmission power variation consists of a square wave signal, defined by two parameters:

- P_T - the amplitude of the signal.
- T_T - the period of the square wave.

.

Likewise, the sampling component of P-DCCA is defined by three parameters:

- T_R - the sample duration.
- S_P - the sample frequency.
- N_R - the number of RSSI samples taken.

The amplitude of the square wave signal, P_T, determines the two transmit powers that are used in P-DCCA transmissions. For example, assuming that the maximum transmit power is used, the minimum for P-DCCA is set $P_T\ dBm$ lower. The relationship between T_R, S_P, and N_R is defined as: $N_R = T_R S_P$.

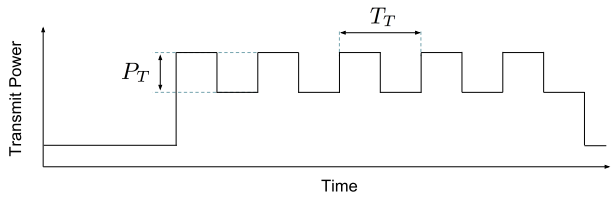
The specifics of the radio hardware must be accounted for when selecting these parameters. So that either extrema is recognisable within the averaged sample window, T_T must not be less than $2A_v$. To be able to distinguish a P-DCCA signal from any other signal with increasing or decreasing signal strength, the recorded sample set must encapsulate at least two extremas of the power variation signal. Thus:

$$2A_v \leq T_T \leq T_R \tag{5.1}$$

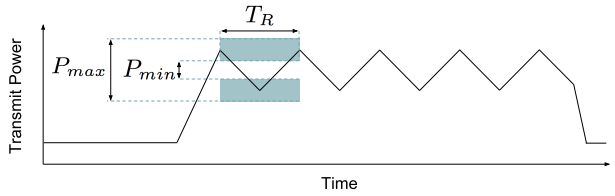
To conserve energy, a minimal sampling duration in a P-DCCA check is desirable, so T_T and T_R can be set to $2A_v$. As consequence, the square wave of the P-DCCA transmitted signal (see figure 5.3a) is perceived as a triangle wave at the receiver, as shown in figure 5.3b. This has the added benefit of keeping the RSSI continuously increasing or decreasing, which simplifies the detection algorithm.

The P-DCCA detection algorithm must be able to reliably detect the waveform as shown in figure 5.3b. In order not to draw resources from the other functions of a sensor node, the algorithm must be minimal in terms of computation time. This is more so the case in WSN MAC protocols that implement a CCA check multiple times in order to receive or send a packet. Likewise, the P-DCCA algorithm must have a small memory footprint in order to run on resource constrained hardware.

The body of a P-DCCA check is illustrated in algorithm 2, which first collects the RSSI trace, then iterates through the sample set to calculate characteristics indicative of P-DCCA. These include the number of extrema, measured amplitude, and change



(a) P-DCCA Transmitter power modulation



(b) P-DCCA Receiver detection of signal

Figure 5.3: P-DCCA operation: transmission and detection

between individual samples. In big-O notation, the running time and memory use are both $O(n)$, where n is the number of samples.

Lines 1-3 collect and compare the RSSI samples - stopping once N_R have been collected, or a sample is below the minimum threshold, τ_{RSSI}. The frequency of this loop must be sufficiently predictable, as measured by S_P. If only a single sample is recorded below τ_{RSSI}, a clear channel is indicated. Therefore, in response to a clear channel, the operation of P-DCCA is no different from standard Energy-Detection CCA. If fewer than N_R samples are detected above the threshold, an inconclusive result is indicated. This is discussed in more detail later.

Lines 5-12 iterate through the sample set to compute the minimum, maximum, and extrema count. This processing is separate to the sampling loop in order to allow the sampling frequency to optimised. As well, the conditional statements herein do not allow for a predictable computation time per iteration.

Lines 13-15 check the amplitude of the signal, and the number of extrema, against predefined thresholds. A signal falling within these thresholds is indicated as a valid P-DCCA modulated signal, otherwise as interference.

Summarised, the thresholds used by this algorithm are discussed below:

1. P_{min} and P_{max} - the maximum and minimum range thresholds, within which the measured range of the sample set must fall between. These effectively measure the

amplitude of the triangle wave. The measured amplitude is subject to variations as consequence of imperfections in the antenna and radio components, and signal propagation in the environment. The thresholds must account for this.

2. P_{Δ} - the maximum allowable absolute difference between adjacent samples.

3. N_E - the maximum number of detected extrema in a sample set. Unlike the signal amplitude, this should not be affected by signal propagation or radio properties.

4. τ_{RSSI} - The minimum RSSI threshold operates identically to the RSSI threshold as used by the standard ED CCA. However, when selecting this threshold, consideration must be given to the variation in signal strength that is part of the P-DCCA design.

1: **for** i = 1 **to** N_R **do**
2: **if** (s[i] = RSSI()) $< \tau_{RSSI}$ **then break**
3: **if** $i = 1$ **then return** CLEAR
4: **if** $i < N_R$ **then return** BUSY_INCONCLUSIVE
5: **for** i=1 **to** $(N_R - 1)$ **do**
6: **if** $|s[i] - s[i+1]| > P_{\delta}$ **then return** BUSY_OTHER
7: **if** $s[i] > s[i+1] \wedge slope \neq SLOPE\_INCREASING$ **then**
8: $slope \leftarrow SLOPE\_INCREASING$
9: $counter \leftarrow counter + 1$
10: **if** $s[i] < s[i+1] \wedge slope \neq SLOPE\_DECREASING$ **then**
11: $slope \leftarrow SLOPE\_DECREASING$
12: $counter \leftarrow counter + 1$
13: **if** $(range(s) < P_{min} \vee (range(s) > P_{max}) \vee (counter > N_E))$ **then**
14: **return** BUSY_OTHER
15: **return** BUSY_154

Algorithm 2: Power Differentiating Clear Channel Assessment (P-DCCA).

5.3.1 P-DCCA Outcome

P-DCCA can have four distinct outcomes:

1. *CLEAR*: indicates that the medium is currently free.

2. *BUSY_PDCCA*: indicates that a transmission with P-DCCA power modulation was detected.

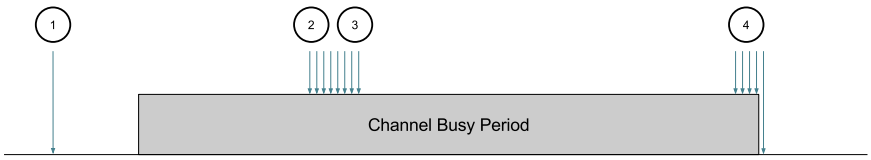


Figure 5.4: P-DCCA outcomes: 1) One sample: channel is clear (*CLEAR*) 2) Complete sample set: channel either *BUSY_PDCCA* or *BUSY_OTHER* 3) Incomplete sample set: channel busy, unknown origin: *BUSY_INCONCLUSIVE*

3. *BUSY_OTHER*: indicates that another signal without P-DCCA was detected, such as WiFi, Bluetooth, or another 802.15.4 device outside the network.

4. *BUSY_INCONCLUSIVE*: indicates that the medium is busy but the channel occupier cannot be determined.

P-DCCA requires N_R RSSI samples, above the threshold, to identify a signal. Therefore, if a P-DCCA check coincides with the end of a busy period, an incomplete RSSI set may be recorded and no source may be definitively identified. This results in a *BUSY_INCONCLUSIVE* result, the interpretation of which is left to the MAC layer. Situations leading to these P-DCCA outcomes are shown in figure 5.4.

The probability of an *INCONCLUSIVE* result, P_I, is inversely proportional to the size of the packet, as given in equation 5.2. $P_{Payload}$ is the size of the packet payload, and P_{Header} is the size of the packet header - include preamble, SFD and length field, both in bytes.

$$P_I(P_{Payload}) = \frac{{S_P}^{-1}(N_R - 1)}{32 \cdot 10^{-6} \cdot (P_{Payload} + P_{Header}) + {S_P}^{-1}(N_R - 1)} \tag{5.2}$$

Node behaviour after obtaining a P-DCCA result is to be decided by upper layers. *CLEAR* may be treated similar to a clear resulting from a normal CCA. *BUSY_PDCCA* would trigger the back-off procedures designed to coordinate competition for the channel among nodes of the same network. The reaction to *BUSY_OTHER* may depend on knowledge of the deployment area. For example, it might be known that other interference is likely to stem from a co-located WiFi network. Likewise, to receive a packet, nodes can only enter the listen phase after a *BUSY_PDCCA* indication.

In case of *BUSY_INCONCLUSIVE*, the decision may depend on worst-case or best-case assumptions. For example, a prudent node listening to the channel for incoming packets may treat this result as normal 802.15.4 traffic, leaving the radio powered on to receive data, while an energy-conscious node may ignore altogether any interference except confirmed 802.15.4 traffic.

Alternatively, *BUSY_INCONCLUSIVE* results can be avoided altogether by stipulating design requirements on the MAC protocol. Knowing the duration between packet transmissions, two P-DCCA checks can be arranged so that at least one will fall within a packet transmission. Thus, inconclusive results can be ignored entirely. This approach ensures that at least one sample set captures a P-DCCA signal, ensuring reliable operation. This is already the case with ContikiMAC, which uses between two and six CCA checks to detect ongoing packet transmissions.

5.3.2 Implementation on CC2420 Transceiver

P-DCCA was implemented on the Moteiv Tmote Sky [She04] hardware. The basic CC2420 radio driver in Contiki implements CCA very simply: checking the radio is powered on then checking the GPIO pin connected to the CCA pin on the CC2420. In the P-DCCA implementation, this was replaced with the algorithm described above. The maximum RSSI register sampling frequency achieved was $33Khz$, which was reduced to $31.2Khz$ in order to coincide with every second symbol, providing more predictable timing. This was calibrated by raising and lowering a GPIO output pin on each loop iteration, and measuring the signal frequency on an external oscilloscope. This code code remained in the CC2420 driver after calibration.

In the CC2420, the average window size, A_V, is $128\mu s$, thus the driver requires upto $128\mu s$ to calculate RSSI and CCA, depending on how long the radio has been powered on, and in receive mode. In the P-DCCA implementation, the driver requires between $256\mu s$ and $384\mu s$, to capture eight RSSI samples in total, the processing time being negligible.

The parameter settings used for the CC2420/Tmote Sky implementation are given in table 5.1. The minimum T_T and T_R are used, based on the CCA average window size.

Parameter	Description	CC2420
S_P	The sample frequency	$31.2Khz$
P_T	Difference between the two used power levels	$5dBm$
T_T	Transmit power variation period	$256\mu s$
T_R	Duration of a P-DCCA check	$256\mu s$
N_R	Number of RSSI samples taken during a P-DCCA check	8
P_{min}	Minimum power range of the P-DCCA sample set	$2dBm$
P_{max}	Maximum power range of the P-DCCA sample set	7 dBm
P_δ	Maximum power difference between two consecutive P-DCCA samples	$4dBm$
N_E	Maximum number of extrema in the P-DCCA sample set	2
τ_{RSSI}	The minimum threshold for each RSSI sample	-75dBm

Table 5.1: P-DCCA parameters and values for CC2420 802.15.4 Transceiver.

Based on the sample frequency, eight samples are recorded per P-DCCA check. A power difference of $P_T = 5dBm$ was used, as this was observed to be the lowest setting yielding good detection accuracy. The transmit power alters between $0dBm$ and $-5dBm$.

To transmit, the standard CC2420 driver strobes the TRX_ON register, which initiates the packet transmission. The driver then waits in a busy-loop until the CC2420 status returns to idle, indicating that the packet has finished transmitting. In the P-DCCA implementation, the body of this loop was utilised instead to vary the output transmission power, with a set period $T_T = 256\mu s$. Consequently, there is no change in the semantics of the radio driver API call to transmit a packet. The P-DCCA function call was measured to take up to $260\mu s$, as opposed to the immediate response with normal CCA; this timing component must be accounted for in the implementation of MAC protocols.

5.4 P-DCCA Detection Evaluation

The DCCA response to interference relies the underlying mechanism (such as T-DCCA and P-DCCA), in order to react correctly and reliably. It must be able to accurately

classify interference and non-interference (WSN communication) accordingly. If this is not the case, DCCA may wrongly interpret WSN data as interference, or vice versa. For example, this could cause a given receiver node to ignore transmitted WSN data, making worse the performance of the network under interference. In either case, it is likely that packet reception or energy efficiency could be adversely affected, even under interference-free conditions.

Therefore, it is the objective of this section to evaluate the outcome of the P-DCCA classification method in two regards. Firstly, in regard to the rate of inconclusive results, as a function of packet size. This is used to evaluate the accuracy of the model described earlier in equation 5.2. Secondly, in regard to the classification accuracy of P-DCCA and T-DCCA, for different interference types. In this case, the classification algorithms of P-DCCA and T-DCCA are dependent on interference type, radio propagation properties, receiver hardware, among other variables. Theoretical and simulation-based models are not able to encompass all of these possible variables. Therefore, an experimental evaluation is required. This must include a transmission type - either interference or non-interference (WSN data) - as input to the DCCA classification algorithm. Therefore, two metrics for measuring classification accuracy are considered for this evaluation:

1. **True Positive (TP) Rate** - the rate of WSN transmission detection by the DCCA algorithm when WSN transmissions are present.

2. **False Positive (FP) Rate** - the rate of WSN transmission detection by the DCCA algorithm in the presence of non-WSN interference.

For this evaluation, the input to the classification mechanism is the independent variable, and must therefore be controlled. In the following sections, accuracy is therefore measured by controlling the interference environment, by injecting either WSN traffic (to measure TP-accuracy), or interference (to measure FP-accuracy). The outcome of both P-DCCA and T-DCCA was then recorded.

To remove the effects of other interference outside of the experiment, all experiments were conducted in environments with few other wireless devices. In each case, this was

verified to confirm that external interference was minimised.

5.4.1 Rate of inconclusive results

In the first experiment, the rate of inconclusive results indicated by P-DCCA was measured, using 802.15.4 packet size as the independent variable. The objective of this experiment was to evaluate the assumptions of the model presented in equation 5.2.

Two Tmote Sky nodes were placed in an unused office: one node acted as the sender, and transmitted packets with the P-DCCA power variation. This experiment measured only the physical and link layers with regard to P-DCCA, and not higher network layers. Therefore, no MAC protocol was used, and packets were sent without any CCA. The second node acted as the receiver, and continuously called the P-DCCA function, as described above. The outcome of the P-DCCA check was transmitted via wired connection to a host computer, where the results were processed.

In this experiment, the RSS of a WSN packet - and consequently the range between transmitter/receiver - is not included as a variable (this is evaluated separately, in section 5.6). Instead, only the timing of P-DCCA packet transmissions and RSSI samples is considered - as in equation 5.2. The RSS must therefore be above the minimum RSSI threshold, τ_{RSSI}. To ensure this, the distance between the two Tmote Sky nodes was $4m$, and the maximum transmission power was used.

The rate of inconclusive results is shown in figure 5.5, alongside the earlier model in equation 5.2. The results show, firstly, that the predicted inconclusive function P_I and the actual recorded inconclusive rate share similar trends: the inconclusive result rate decreases as packet size increases; and this behaviour appears to be asymptotic, as per equation 5.2.

The results do not closely follow the predicted model, however. For packet sizes in the range $[40, 80]$, the model underestimates the recorded inconclusive error rate; the average deviation from P_I is 1.4%, within this range. The largest deviation from the model occurs for packet size 80 bytes, where the predicted and recorded inconclusive rates are 7.5% and 9% respectively. In applying this model, this may consequently underestimate

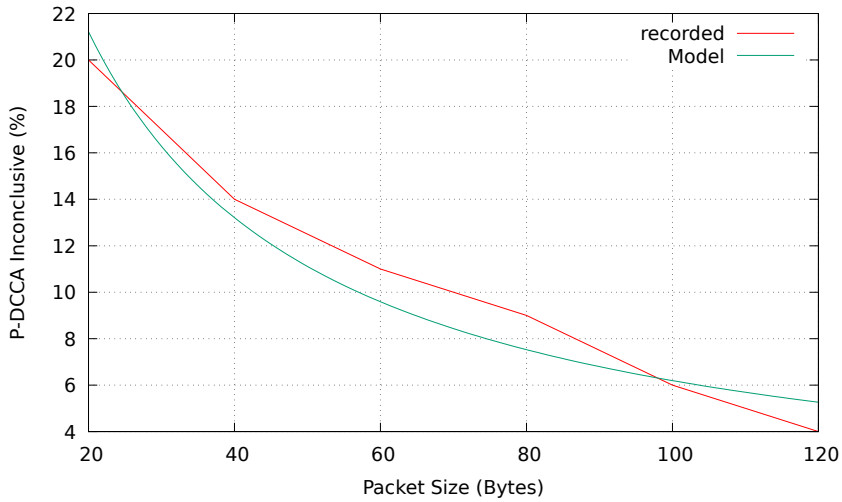


Figure 5.5: Rate of inconclusive classification in P-DCCA as function of packet size.

the rate of inconclusive results. Therefore, correct classification of WSN data may be reduced below that predicted by the model, which may subsequently impair operation of DCCA in the MAC protocol.

The model and recorded results cross in two locations: the former overestimates the inconclusive result rate for packet size below 40 bytes, and above 80 bytes. The model therefore presents an upper-bound of inconclusive result rate, in these cases.

These deviations may be due to timing variations in the implementation of the algorithm which are not accounted for in the model. Alternatively, this may be due to other interference in the environment. Observing the graph in figure 5.5, the deviation from the model seems to increase as a function of packet size. This may be because the probability of observing an inconclusive result decreases with packet size, and therefore variation within the results increases. A longer experiment duration for each packet size may have mitigated this.

In this experiment the accuracy of the predicted inconclusive function, P_I, could not be verified. However, based on the similarity between trends in figure 5.5, the underlying assumptions used to derive equation 5.2 are supported. Namely, that P_I stems from the ratio of two durations in receiving a packet: the duration susceptible to inconclusive detection ($S_P^{-1}(N_R - 1)$), and the duration of transmitting the packet payload. Therefore, the objective of this experiment has been met. The model predicts

an asymptotic relationship between packet size and the inconclusive result probability: as packet size increases, the rate of inconclusive results approaches zero. This is seen in figure 5.5.

The standard CCA mechanism is not an instantaneous calculation, calculating the average channel energy over $128\mu s$, although a signal only needs to occupy this long enough to raise the calculated average above the threshold. Thus, the standard energy-detection CCA similarly requires the transceiver to have captured a minimum duration of the transmitted signal.

5.4.2 True Positive (TP) Rate

The objective of this experiment is to evaluate the TP-accuracy of the P-DCCA and T-DCCA classification algorithms. This measures the ability of the classification algorithm to identify correctly WSN traffic, such as from another WSN node, correctly.

For this experiment, two Tmote Sky nodes (transmitter and receiver) were placed in an unused computer laboratory during quiet office hours. The TP accuracy of the receiver node was measured by its ability to detect these packets sent by the transmitter. While the objective of this experiment focuses only on the accuracy of the classification algorithm, this hardware is typical of WSNs. Alternatively, more specialised channel sampling hardware could have been used to achieve greater sample resolution and frequency. However, this is not realistic in typical deployments.

RSS and also range between transmitter/receiver was not included as a variable in this experiment. Instead all DCCA variants were evaluated under the same condition: incoming packet RSS is reliably above the RSSI threshold. Consequently, the range between devices in this section was set as $4m$, and the maximum transmission power was used. This is based on the assumption that variations in RSS over time - which both algorithms use to classify interference - is independent of range between transmitter and receiver.

The experiment was repeated five times and in each iteration the P-DCCA and T-DCCA variants were tested separately. This duration was chosen based on the assump-

tion that there is no network or initialisation overhead implemented in the WSN nodes. Hence there is no initialisation period at the start of each experiment to account for: the results are immediately valid. Since the interference is strictly controlled, no temporal variance within the duration of each experiment is expected, therefore allowing for a shorter test duration. The nodes were reprogrammed over USB by the host computer to change the DCCA variant as required.

To evaluate T-DCCA, the offline implementation of ZiSense described in Section 5.2 is used. The duration of a T-DCCA check and its relation to the LPL transmission timing ensures that each T-DCCA check can obtain sufficient data to identify the ongoing WSN transmission.

To evaluate P-DCCA, it is ensured that the P-DCCA check is carried out while a packet transmission is ongoing. To achieve this, transmitter and receiver node are synchronised via a cable connection. Thus, T-DCCA and P-DCCA have both the chance to correctly identify an ongoing transmission within the DCCA check. However, it has to be noted that both DCCA checks have very different time scales ($2.9ms$ compared to $256\mu s$).

To achieve this synchronisation, a wire was run from the transmitter to the receiver, connecting GPIO pins on both nodes. The transmitter set the pin high immediately before transmitting, and low immediately after, for every packet. The receiver node then sampled the pin status before and after each P-DCCA function call, and communicated both to the host computer as well as the P-DCCA return value. To measure TP accuracy, only P-DCCA calls that completely coincided with a packet were included.

The results are shown in figure 5.6. The results show, firstly, that P-DCCA achieves higher TP accuracy than all ZiSense classifications. ZiSense-2 measured the highest TP accuracy, compared to ZiSense-1 and ZiSense-C4.5. P-DCCA TP-accuracy is 88%, while ZiSense-1, ZiSense-2, and ZiSense-C4.5 have 78%, 86% and 78% respectively.

Figure 5.6 also shows a larger standard deviation for P-DCCA, compared to ZiSense. This may be due to differences in the sample size between ZiSense and P-DCCA evaluation methods. ZiSense was evaluated offline for all contiguous sub-samples within

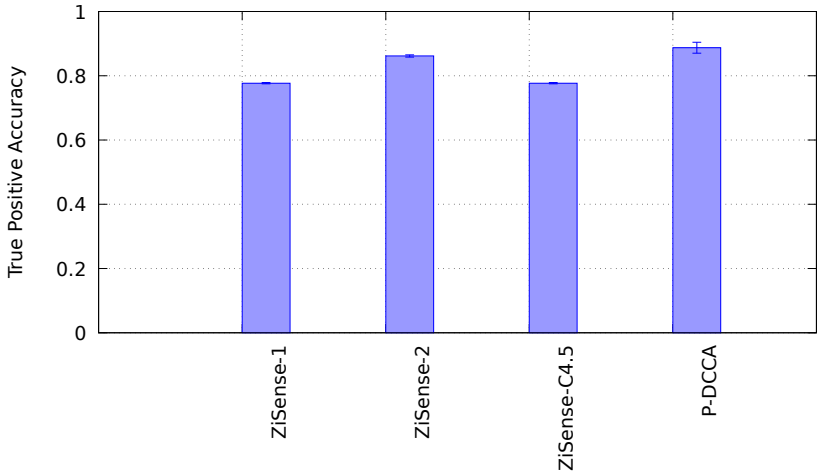


Figure 5.6: True Positive rate of P-DCCA and ZiSense.

the recorded RSSI trace, while P-DCCA was evaluated via an online implementation. This experiment shows that using P-DCCA, high classification accuracy can be achieved despite a comparably shorter sampling duration ($2.9ms$ compared to $256\mu s$).

5.4.3 False Positive (FP) Rate

The objective of this experiment was to evaluate the FP-accuracy of the DCCA classification algorithm. This measured the ability of the classification algorithm to identify correctly non-WSN interference, such as WiFi. For this experiment, two Tmote Sky nodes were programmed to act as P-DCCA and T-DCCA receivers, and placed in an unused computer laboratory. As in the previous experiment, this hardware was preferred as is it more representative of actual an WSN deployment. No MAC protocol was used in this experiment, since no data is communicated and energy consumption is not being evaluated.

As previously, the environmental interference was strictly controlled, therefore allowing for a short test duration. The experiment took place during quiet office hours, with minimal external Radio Frequency (RF) activity. The interference sources in this experiment are less predictable in terms of behaviour due to the complexity of the underlying state machine. Therefore, the number of iterations in this experiment was increased to ten, each iteration taking five minutes.

Five types of $2.4Ghz$ interference were included in this experiment. To best represent typical interference sources, the configuration of each interferer was left at the default setting, where applicable. As previously, RSS of interference is not included in this experiment. Therefore, the interference sources - described below - were placed $5m$ from each WSN node.

- **WiFi**: An 802.11 AP and station were placed at opposite corners of the room. Both were connected to a host computer over a wired connection, which acted as the control channel for the experiment. Interference was generated using the D-ITG tool [AGE+04]. 1000-byte UDP packets were transmitted at a uniformly random rate as specified. This was done to ensure there was no unintended synchronisation between the WiFi data packets and the DCCA sampling, which could skew the results. 802.11 b, g, and n variants were used in the experiments, as configured by the host computer. These were chosen as the most common 802.11 variants in typical $2.4Ghz$ deployments.

- **Bluetooth**: Bluetooth traffic was generated by sending a large file between two devices: a MacBook Pro laptop computer and a Google Nexus 5 Android smartphone.

- **Microwave Oven**: MWO interference was generated using a household microwave oven, heating $500ml$ of water in a pyrex bowl at 800 Watts.

Both nodes continuously called the DCCA function, the outcome of the detection mechanisms recorded on the host computer. A false positive result is recorded whenever an 802.15.4 packet is detected.

The results are shown in Figure 5.7. The results show, firstly, that for all interference sources, P-DCCA recorded a lower FP rate than all ZiSense variants. ZiSense-2 recorded the highest FP rate for all interference sources, while in most cases ZiSense-1 and ZiSense-C4.5 were similar.

Secondly, figure 5.7 shows that 802.11b is the most susceptible to false positives, for all DCCA implementations. This is particularly the case for ZiSense implementations,

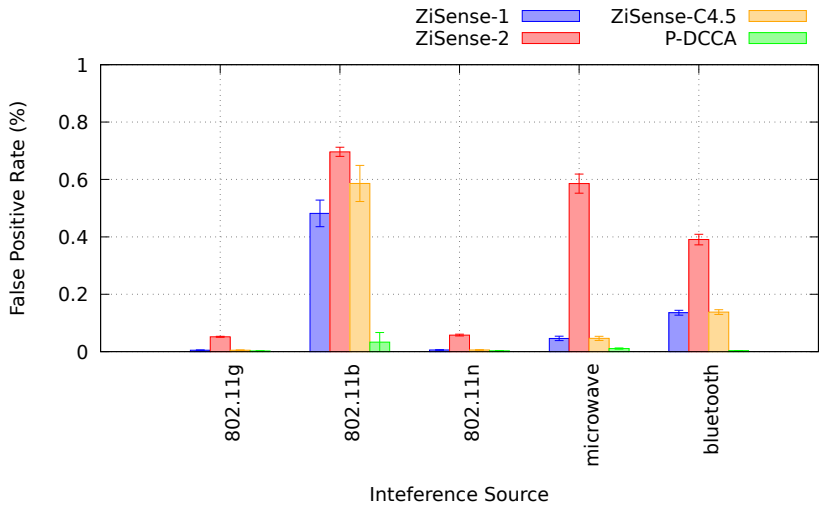


Figure 5.7: False Positive rate for P-DCCA and ZiSense.

however. This is due to the closer observable similarity between 802.15.4 and 802.11b signals: the slower transmit rate and DSSS modulation causes the packet duration and PAPR to fall within ZiSense thresholds.

Conversely, 802.11g and n have the lowest FP rate across all DCCA approaches. This is because, in both cases, the transmission duration is well below the threshold for P-DCCA and T-DCCA. Likewise, 802.11g and n use OFDM modulation, which has different RSSI trace features, compared to DSSS [ZCW+14].

For the MWO interference, the false positives in P-DCCA maybe caused by the inherent power oscillation of MWO interference being perceived as a P-DCCA transmission. Similarly, false positives in T-DCCA may be the result of MWO signals falling within the defined thresholds.

This experiment has shown that P-DCCA is more accurate in terms of TP and FP accuracy, compared to ZiSense, in these conditions. This is beneficial, since DCCA requires correct classification of interference signals, in order to mitigate the effects of CTI. However, it has to be noted that P-DCCA can return *BUSY_INCONCLUSIVE*. In this case an FP is avoided but not enough information is available to discern the interferer type. Inconclusive results can be mitigated by applying multiple P-DCCA checks.

5.4.4 Discussion

The TP and FP accuracy of ZiSense shown in figure 5.6 and figure 5.7 is less than reported in [ZCW+14], wherein much higher accuracy was measured. However, these results bare similar trends relative to the author's findings in [ZCW+14]. ZiSense-2 is the most susceptible to FP error, but achieves the highest TP accuracy. Conversely, ZiSense-1 and ZiSense-C4.5 have better FP rate for a reduced TP rate.

This discrepancy could be due to differences in the WSN hardware, such as RSSI accuracy, used in the evaluation. Alternatively, this could stem from environmental factors, such as other interference or RF propagation effects. To the best of the author's knowledge, this implementation of ZiSense matches the original, and the same method was used to evaluate accuracy. Given that the similar trends between ZiSense implementations have also been observed here, it is likely that the difference in absolute accuracy stems from environmental and hardware differences.

The author's original evaluation did not factor 802.11b interference, a standard that is now considered obsolete compared to later g, and n, standards. Despite this, the rate selection algorithms of many 802.11 devices - which is manufacturer specific - may fall back to 802.11b PHY in adverse channel conditions. This was observed to be the case where an AP is vying for channel access under heavy 802.15.4 interference.

In implementing DCCA in a WSN MAC protocol, the implications of a high FP rate must be considered. False wake-ups - where non-802.15.4 signals are classified as incoming WSN traffic, cause increased idle listening and hamper energy efficient operation. When using DCCA, the probability of a false wake-up occurring is a function of the FP rate. $BUSY\_INCONCLUSIVE$ results can occur as well but can be interpreted as $BUSY\_OTHER$ to improve energy efficiency.

In P-DCCA, the sampling duration is variable: samples are recorded only whilst the RSSI is above the threshold. Conversely, the ZiSense sampling duration is fixed and much longer ($2.9ms$ for ZiSense compared to a maximum P-DCCA duration of $256\mu s$). Furthermore, it has been shown that P-DCCA has better TP and FP rates compared to ZiSense.

5.5 Energy Evaluation

P-DCCA, T-DCCA, and standard CCA implementations all use the radio differently, have different listen durations, and therefore have different energy consumption profiles during a channel check. This affects the energy efficiency and battery life of the node - possibly offsetting the benefits of using DCCA in practice. Therefore, it is necessary to evaluate and compare these in terms of energy consumption. The objective of this evaluation is also to gain further insight into the underlying channel sensing mechanisms of each DCCA method.

In this section, a theoretical model of the energy consumption incurred by T-DCCA, P-DCCA, and standard CCA is produced. A theoretical evaluation is chosen over an empirical evaluation due to lack of an available ZiSense implementation for the Tmote Sky. Also, the channel sensing behaviour of P-DCCA is variable depending on the nature and type of local interference; hence a theoretical model is beneficial because this behaviour can be deliberately isolated. Likewise, this evaluation method is immune to differences between MAC and network protocols of different DCCA implementations. ZiSense is again used as a comparable reference for T-DCCA, as described in [ZCW+14]. The P-DCCA implementation described in Section 5.3 is assumed here. The idle listening duration per channel check is modelled - since this is the greatest source of energy consumption in low-traffic deployments.

Any transceiver activity within the P-DCCA mechanism is assumed to have the same energy consumption; energy consumption during transceiver start-up and during RSSI sampling is the same. This is a realistic approximation for transceivers such as the CC2420 commonly used. Thus, energy consumption of a P-DCCA check depends on the transceiver-on duration, T, which is modelled.

The behaviour of the P-DCCA algorithm described depends on the interference encountered, which must be provided as input to this model. The probability of finding the channel busy at any instant is represented as p; when busy, the channel is then occupied by an interference signal with fixed duration t. For example, when considering a WiFi interferer an intensity of p and an average WiFi packet length of t is modelled.

This model of interference is less rigorous than other statistical models and approaches. However it simplifies the mathematical models that follow, and is sufficient to derive an upper-bound for T to enable comparison.

The expected duration T of a P-DCCA check (shown in Equation 5.3) is the sum of the transceiver start up time, T_{st}; the duration of the first RSSI sample, which is always taken; and the time taken to carry out the sequence of $(N_R - 1)$ RSSI samples as described in Algorithm 2. The latter stems from two possible outcomes. Firstly, T_A models the case where the P-DCCA check starts sooner than $(N_R - 1)$ RSSI samples before the end of the interference. Otherwise, fewer samples are taken (T_B).

$$T(t, p) = T_{st} + T_{RSSI} + p \cdot \big(T_A(t) + T_B(t)\big) \tag{5.3}$$

T_A is given in Equation 5.4, and is the product of the total duration of all RSSI samples: $T_{RSSI} \cdot (N_R - 1)$, and the probability.

$$T_A(t) = \Big(T_{RSSI} \cdot (N_R - 1)\Big) \cdot \Big(\frac{t - (N_R - 1) \cdot T_{RSSI}}{t}\Big) \tag{5.4}$$

The remaining interference duration, modelled by T_B, has duration $t - T_{RSSI} \cdot (N_R - 2)$; this is considered as $(N_R - 2)$ discrete intervals. The probability of each is given by T_{RSSI}/t, and the duration of the subsequent n samples is given by $n \cdot T_{RSSI}$. Therefore, T_B is the sum expected duration of these $(N_R - 2)$ intervals.

$$T_B = \sum_{n=1}^{N_R - 2} n \cdot T_{RSSI} \cdot \Big(\frac{T_{RSSI}}{t}\Big) \tag{5.5}$$

This model assumes firstly that $t \geq N \cdot T_{RSSI}$: that the duration of the interference signal is greater than the number of RSSI checks in P-DCCA. This is not necessarily true for all types of interference, in which case T represents an upper-bound. Secondly, the model assumes that p and t are independent variables.

As discussed earlier, implementing P-DCCA in a MAC Protocol requires consideration of the packet timing: a minimum number, n, P-DCCA checks are required to ensure reliable detection. The time between P-DCCA checks must be shorter than the minimum

packet length, and greater than the inter-packet spacing.

The ZiSense implementation was originally based and evaluated on the TinyOS LPL MAC protocol, whose inter-packet spacing is $2.9ms$. Assuming an arbitrary minimum packet size of at least $500\mu s$, $n = 6$ checks spaced $500\mu s$ apart, over $3ms$, is sufficient for implementing P-DCCA.

Based on this, the radio-on time per DCCA check for ZiSense and P-DCCA are shown in figure 5.8, for channel conditions $p = 0.1, 0.25, 0.5$. For comparison, the radio-on time for plain CCA check is shown also (negating false wakeups). However, plain CCA would not be able to infer the nature of the channel occupier and any observation of interference would lead to a false wake-up contributing significantly to energy consumption; this is not considered here, and plain CCA is included only as a reference point.

This model shows the relationship between environmental interference and the idle listening in P-DCCA. As shown, P-DCCA requires less radio-on time to sample the channel than ZiSense. Under quiet channel conditions, the idle listening duration, per wakeup, approaches that of plain CCA. As the interference intensity increases, the idle listening of P-DCCA increases asymptotically, and does not exceed $1.5ms$, for these parameters. The reason for this behaviour is apparent considering T_A and T_B. The former can be rearranged as in equation 5.6.

$$T_A(t) = \Big(T_{RSSI} \cdot (N_R - 1)\Big) \cdot \Big(1 - \frac{(N_R - 1) \cdot T_{RSSI}}{t}\Big) \tag{5.6}$$

As t increases, the second term in equation 5.6 approaches one, while T_B approaches zero. Consequently, the behaviour as $t \to \infty$ is shown in equations 5.7 and 5.8 respectively.

$$\lim_{t\to\infty} T_A(t) = T_{RSSI} \cdot (N_R - 1) \tag{5.7}$$

$$\lim_{t\to\infty} T_B(t) = 0 \tag{5.8}$$

Hence, this leads to the observed asymptotic behaviour. By contrast, the idle listening of ZiSense remains fixed under all conditions, at $2.9ms$.

This estimation incorporates only the cost of listening to the channel. The response to

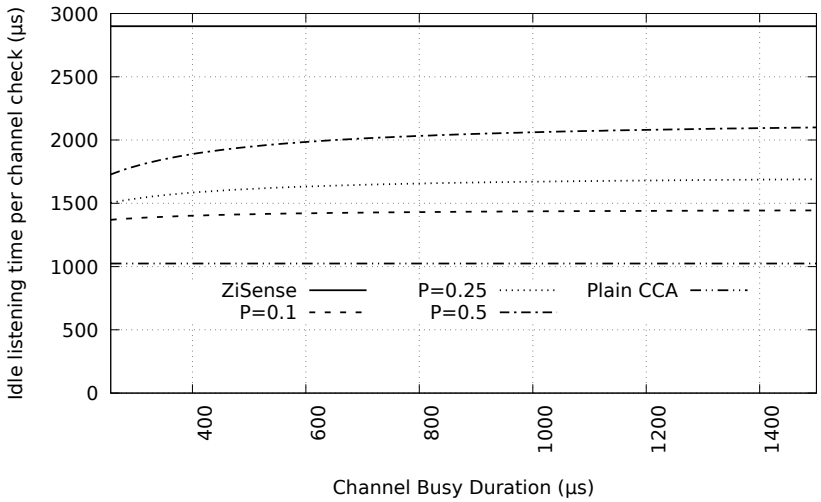


Figure 5.8: Modelled energy consumption of P-DCCA and T-DCCA under interference.

false positives - which is MAC protocol-specific - is not accounted for, however P-DCCA was shown in Section 5.4.3 to have greater detection accuracy compared to ZiSense. Also, the signal processing cost - negligible for P-DCCA - is excluded.

In this section, the energy consumption of P-DCCA and ZiSense implementations have been compared theoretically. P-DCCA has been shown to have a lower idle listening cost, leading to better energy efficiency. While this model of energy consumption is not supported empirically, it is sufficient to gain further insight into the effect of interference on T-DCCA and P-DCCA mechanisms. Namely, that the energy efficiency of P-DCCA is determined by the interference, and idle listening will be minimised in quiet environments. This relationship between interference and idle listening is asymptotic, and hence idle listening cannot exceed this limit. Consequently, regardless of the intensity of interference, the idle listening of P-DCCA is expected to be lower compared to T-DCCA for these parameters. This is observable in figure 5.8. The energy efficiency benefits to WSNs of P-DCCA are evaluated in real experiments in section 5.7.

5.6 Range Evaluation

The transmission power variation used by P-DCCA reduces the signal strength, and consequently the SNR of the received signal. As consequence, this may prevent the

802.15.4 receiver from being able to correctly decode a received signal. This may lead to increased bit- and packet-errors for a given link. SNR is a function of the distance from the transmitter to the receiver, and therefore the maximum range for P-DCCA links may be impaired by P-DCCA power variation. The objective of this section is therefore to gain insight into, and to evaluate, the effect of P-DCCA transmission power variation on link quality.

This evaluation is twofold. Firstly, a theoretical model of packet-error rate is produced, based on previous work by Shin et al [SCPK05]. This takes as input the distance between transmitter and receiver, d, and the transmit power variation setting, P_T, in order to estimate the packet error rate, P. Secondly, an experimental evaluation is described to complement the theoretical model. This approach allows firstly external factors - such as environmental conditions and hardware imperfections - to be excluded in the theoretical model, in order to isolate and study the input variables. It also provides greater insight into how the P-DCCA power modulation affects link quality. The experimental evaluation then allows this model to be validated empirically, and also provide a more rigorous range evaluation under realistic conditions.

The packet error rate, P, is modelled as a function of distance and the P-DCCA transmit power variation setting, P_T. P is described later as a function of SNR, which is determined by the RSS of the received signal. For a non-P-DCCA transmission, this is defined in equation 5.9 as a function of the transmission power, P_{TX}, and distance from the sender, d.

$$R(P_{TX}, d) = P_{TX} - L(d) \tag{5.9}$$

The path loss function, $L(d)$, is independent of the signal transmission power. Equation 5.10 expresses the path loss, L, based on a log distance model, taken from [SCPK05].

$$L(d) = \begin{cases} 40.2 + 20\log_{10}(d), & \text{if } d \leq 8 \\ 58.5 + 33\log_{10}(\frac{d}{8}), & \text{if } d > 8 \end{cases} \tag{5.10}$$

The Bit Error Rate (BER) is modelled as B, and for an Additive White Guassian Noise (AWGN) channel can be modelled as in equation 5.11 [SCPK05].

$$B = Q\Big(\frac{2SNR}{0.083}\Big) \tag{5.11}$$

Where SNR is defined as ratio between the incoming signal strength, R, and the noise power, P_N. The noise power is the sum of all other signals, including thermal noise, hardware noise, and interference from other devices.

$Q(x)$ is the Gaussian error integral, which can be approximated as:

$$Q(x) = \frac{e^{-(x^2/2)}}{1.64x + \sqrt{0.76x^2 + 4}} \tag{5.12}$$

For a uniform signal power, the packet reception rate, P, can be modelled from B by the equation 5.14.

$$P(s) = (1 - B)^{8s} \tag{5.13}$$

Then, after incorporating 5.11 and 5.12, P becomes:

$$P(d, s) = \left(1 - Q\Big(\frac{2\frac{R(P_{TX},d)}{P_N}}{0.83}\Big)\right)^{8s} \tag{5.14}$$

The function P, described in equation 5.14, predicts the PRR as a function of distance, for a standard 802.15.4 transmission. Here, s is the size of the packet to be transmitted, in bytes, including the packet header, and d is the distance between the two nodes. The remaining variables, P_{TX}, and P_N can be fixed.

The model for a P-DCCA transmission is similarly defined in equation 5.15, which is the product of P, for both packet components as transmitted by the P-DCCA square wave. Since the path loss experienced by a signal is independent of the transmission power, the received signal strength of the packet, transmitted at the P-DCCA lower transmit power, is defined as $R(P_{TX} - P_T, d)$.

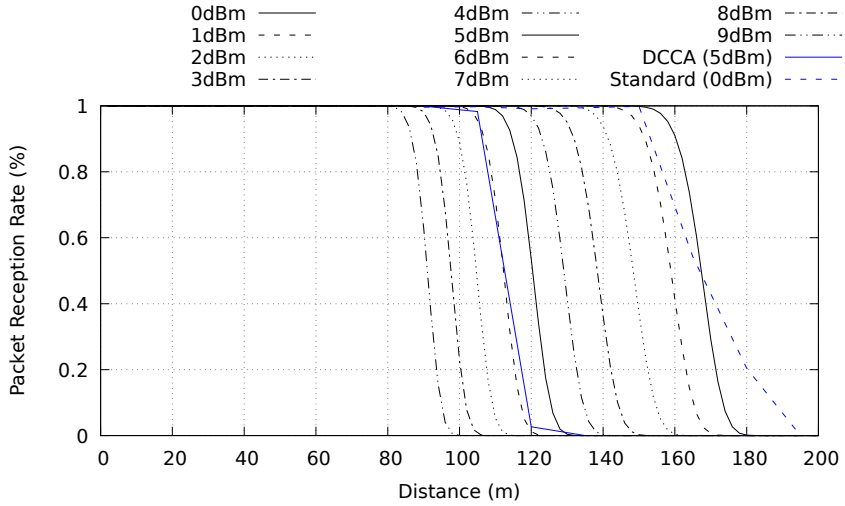


Figure 5.9: PRR of P-DCCA as function of distance between nodes.

$$P_{PDCCA}(d,s) = \left(1 - Q\left(\frac{2^{\frac{R(P_{TX},d)}{P_N}}}{0.83}\right)\right)^{4s} \left(1 - Q\left(\frac{2^{\frac{R(P_{TX}-P_T,d)}{P_N}}}{0.83}\right)\right)^{4s} \tag{5.15}$$

P_{PDCCA} is plotted in figure 5.9, for the distance $\{d : 0 \leq d \leq 200\}$, for values of P_T in the range [0,9]; $P_T = 0$ is uniform transmission. The packet size, s, is set to 90-bytes. The model shows that for various P_T, link reliability exhibits the same behaviour: P is stable to a point, where it drops quickly and becomes unusable. P-DCCA affects the distance at which this occurs, and hence affects the maximum communicable distance of the link. For example, P is negligibly degraded for $P_T = 5$ compared to uniform transmission power, up to $109m$.

This simplistic model firstly does not include the effects of environment properties, such as humidity, air pressure, hardware characteristics, antenna design and multi-path fading. These can significantly affect RF propagation characteristics. Secondly, this model of P-DCCA assumes that the two terms in 5.15 are independent, which may not be the case.

An experiment was run to evaluate the accuracy of this model. Two telosB nodes were configured as sender and receiver; the former was programmed to transmit 90 byte packets at a rate of 16 packets/second; the latter logged each packet received to a buffer

in RAM. The choice of packet size is arbitrary and - as suggested by the exponential nature of the model presented - has little effect on the packet error rate. In this case, 90 bytes is chosen since this is typical of WSN applications based on a TCP/IP and IPv6 architecture.

The experiment took place outside in an empty sports field, absent of any wireless electronic devices in order to minimise interference. The maximum range of P-DCCA and standard 802.15.4 transmissions was previously established to be approximately $90m$ and $180m$ respectively. Hence, the distance between the nodes in this experiment was varied between $90m$ and $180m$, in increments of $15m$. Both nodes were elevated $75cm$ from the ground to ensure line of sight. This range evaluation is not representative of all possible deployments: indoor environments, for example, are subject to RF propagation effects.

Due to the absence of other interference sources, there was little variation in this experiment. Therefore, packet delivery rates were recorded for three minutes, for each distance. For each increment, the packet reception rate - as measured at the receiver, and not considering acknowledgements - was calculated. The transmitter node did not implement CCA checks or retransmissions, and neither node duty cycled the radio.

Two configurations of the P-DCCA transmit function were used, $P_T = 0$ - which is the uniform transmission power, as standard, and $P_T = 5$ - the setting used in the CC2420 implementation in this thesis.

The results of the experiment and the model of P are shown in figure 5.9. A number of observations can be drawn from these results. Firstly, all values of P_T and the recorded results exhibit the same behaviour: PRR stays close to 100% until a threshold is met, after which PRR drops to 0%. From the model, this threshold - which is termed the maximum link range - is inversely proportional to P_T. For example, this threshold is approximately $80m$ for $P_T = 9dBm$.

Secondly, for $P_T = 5dBm$, the recorded PRR closely follows the model: close to 100% PRR is recorded for distances up to $105m$, beyond which no packets are received. This is not the case for $P_T = 0dBm$, where close to 100% PRR was measured in the experiment,

and predicted by the model, up to $105m$. Beyond this, the model and the experiment results deviate however, as the experiment observed above 0% PRR up to $200m$. The latter represents standard 802.15.4 transmission, without P-DCCA power modulation. Consequently, the model in equation 5.15 presents a lower bound of PRR as a function of distance. Therefore, the range impairment of P-DCCA, compared to standard 802.15.4, may be greater than that predicted based on this model.

Discrepancies between the model and the experiment results may stem, firstly, from environmental factors not included in the model. These include humidity, hardware properties, battery voltage and radio propagation effects. Secondly, the model separates the packet into two components, and calculates the PRR of each as a function of transmission power. These are assumed independent, and their product is used to derive the PRR of the P-DCCA transmission. This assumption was not verified by the experiment results.

The accuracy of the model is not supported by the results in figure 5.9. However, the model and experiment results share similarities, and therefore some of the insights gained from the model are supported. Firstly, that below the maximum link range, the PRR of P-DCCA links is unaffected compared to standard 802.15.4. Consequently, only the maximum link range is affected by P-DCCA. Secondly, that maximum link range of P-DCCA is determined by P_T. Consequently, implementations of P-DCCA should minimise this as much as possible, without reducing classification accuracy.

Based on this model, the maximum link range is defined as the distance, d, at which PRR drops below 99%. In many routing protocols, links which do not meet this threshold are avoided, and hence, such links are a requirement of a stable routing topology. For values of P_T in the range [0,9], the maximum link range is plotted in figure 5.10. Here, a linear relation between P_T and link range is apparent.

As discussed in Section 5.1.2, the reduced link performance is the main drawback of P-DCCA. In a large, multi-hop network, more hops may be needed to communicate, increasing energy consumption and offsetting the benefits of P-DCCA. Otherwise, if nodes are too sparsely deployed, P-DCCA may fragment the network and prevent

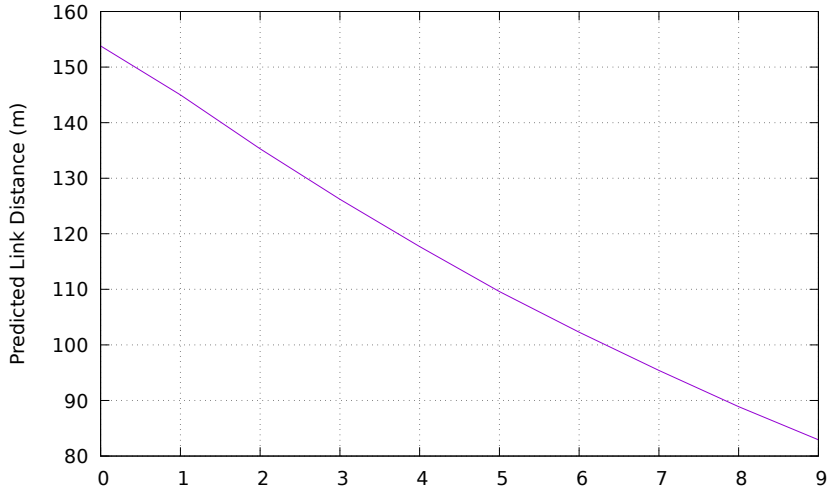


Figure 5.10: Modelled Link Distance of P-DCCA for various P_T

communication entirely. P-DCCA may therefore be better suited to either smaller networks, with few nodes, or large, multi-hop networks, with many possible links.

However, when considering the design of sender-initiated asynchronous MAC protocols, it becomes apparent that the link performance is not the only parameter determining maximum link distance. Receivers implementing these MAC protocols periodically sample the channel energy using CCA checks: if the set CCA threshold is exceeded, an incoming packet is inferred, and the radio remains powered on to receive a packet. Otherwise, the radio is powered off, until the next wakeup sequence. Therefore, for a packet to be received it is necessary that the RSS exceeds the CCA threshold. Since the 802.15.4 PHY is designed to operate even at low SNR levels, it is plausible that this may be the limiting factor of maximum link range in sensor networks.

For example, for the maximum link distance distance of a uniform-transmission power, the received signal strength modelled in figure 5.9 is -100.1dBm, far below the default CCA threshold. Further, it would be counter-intuitive to set the CCA threshold so close to the noise floor - as receivers would not be able to reliably distinguish incoming signals from the noise floor.

5.7 Application Evaluation

In section 5.4, the TP and FP classification accuracy of the P-DCCA detection mechanism was evaluated and compared against T-DCCA. This was done in the absence of any MAC protocol or DCCA response policy. It was shown that while P-DCCA is able to provide accurate (subjectively speaking, and compared to current state-of-the-art alternatives) classification of interference, false-positives and false-negatives are still possible. This may have adverse consequences on a WSN. Similarly, the benefits of employing DCCA in a WSN should be evaluated empirically. Therefore, in this section, it is the objective to evaluate how a DCCA implementation and simple interferer response policies affect link performance and energy efficiency of a WSN under interference.

P-DCCA was implemented in ContikiMAC, alongside simple interferer-detection policies. ContikiMAC was chosen since this represents the current state-of-the-art among WSN MAC protocols, and is a common choice WSN deployments. ContikiMAC is similar in operation to other duty-cycled WSN MAC protocols that rely on CCA to detect incoming traffic and prevent collisions, and hence these findings are assumed to be similar to other MAC protocols in the WSN domain.

The experiments in this section include firstly, small scale experiments to precisely measure link performance and energy efficiency on individual links - absent from the effects of any network or routing protocols; secondly, large scale WSN deployments to evaluate, also, the compounded effect when network and routing are also included. The experiments in this section take place under controlled interference - to measure performance under known channel conditions, and uncontrolled interference - to measure performance under realistic conditions.

5.7.1 ContikiMAC implementation

To measure the effects of P-DCCA on a WSN in the presence of interference, two variations of the standard ContikiMAC protocol were included in all experiments: standard ContikiMAC - to provide a baseline comparison; and ContikiMAC with P-DCCA - to measure the improvement yielded by DCCA.

For the latter, deliberately simple response policies were implemented in order to evaluate the DCCA concept. In the future, more complex policies and mitigation strategies may follow from this. The DCCA outcome is used in both transmitter and receiver modes:

- **Transmitter** Collisions with other WSN traffic during the transmission sequence are followed by the familiar deferral and back off strategy. However, collisions with other interference sources are ignored. This is to prevent transmitters from aborting a packet transmission which may otherwise be received, while still being able to arbitrate channel access amongst WSN nodes.

- **Receiver** During the receiver wakeup sequence, incoming data is only inferred after detecting WSN traffic. All other interference is ignored, and treated the same as a clear channel. This is to prevent ContikiMAC from falsely inferring incoming data from spurious interference on the channel.

The standard transmission policy of ContikiMAC, upon detecting a collision, is to abort the transmission sequence and initiate a random back off. This is sufficient to arbitrate access to the channel between multiple competing WSN nodes.

In cases of collisions with other RF devices which follow different collision policies with vastly different timescales, this approach is likely to be suboptimal - as channel arbitration is unevenly weighted between competing devices. Consequently, this motivates the transmission policy described above: interference outside the WSN is ignored, and packet transmission persists until an acknowledgement is received, or times out. The receiver policy is intended to reduce false wakeups, and preserve energy efficiency of ContikiMAC under interference.

As discussed in Chapter 5, inconclusive P-DCCA results - occurring when an incomplete RSSI sample set is recorded - are ignored in this implementation. Instead, P-DCCA checks take place in pairs, and packet transmissions are predictably timed to ensure that at least one P-DCCA check will coincide with a packet and record a full sample set. This approach is identical to the standard ContikiMAC mechanism.

To evaluate ContikiMAC with, and without P-DCCA, two metrics are used throughout the evaluation:

- **PRR** To evaluate the effect of DCCA on link quality, Packet Reception Rate (PRR) is used. This is a reflection of the link performance, and is the ratio of successful packet transmissions (where an acknowledgement is received by the sender) to total attempts. This is chosen because, from this, the effects on other components of the WSN, such as routing protocols and user applications, can be broadly extrapolated. For example, increased PRR is likely to correlate to improved performance of data collection applications. This is not necessarily the case with other metrics.

- **Radio-on time per packet received** This metric is used to measure the energy efficiency of ContikiMAC under interference. This metric is used since it encompasses the energy expenditure of transmitting, and receiving packets, including the idle listening cost. As false wakeups and collisions necessitating retransmissions increase, so does this metric.

5.7.2 Controlled interference evaluation

In order to evaluate the hypothesis that the DCCA response policies described will improve performance under interference, the objective of the first experiment was to evaluate the link performance and energy efficiency of ContikiMAC in controlled interference conditions.

Two Tmote Sky nodes were placed in opposite corners in an unused office testbed. The effect of signal RSS - which is determined by transmission power and range, of both WSN traffic and interference, was not included in this experiment. In both cases, RSS was intended to be reliably above the CCA threshold. Therefore, the distance between the transmitter and receiver was $6m$, and the maximum transmission power was used.

In the same office, an 802.11g access point and station were placed in opposite corners, and were connected to the host computer over a wired connection. Interference was generated using the D-ITG traffic generation tool [AGE+04], sending UDP traffic. In order to reduce the effect of other interference from other sources within the building, the

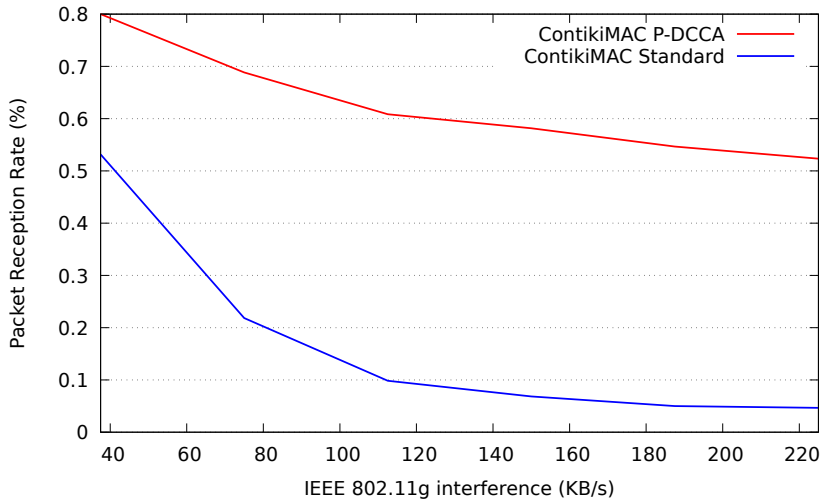
experiment took place during quiet office hours. Likewise, WiFi channel 11 and 802.15.4 channel 22, which were found to be the quietest overlapping channels in the environment, were used for the experiment.

90-byte packets were sent at a rate of 4 packets per second, and sent without retransmissions. This packet size is used to represent a typical WSN data collection application. PRR and radio-on time per packet transmitted - as measured at the receiver, were recorded throughout the experiment. No routing protocol was employed, and therefore the network incurred no initialisation delay. Each interference level was therefore run for five minutes.

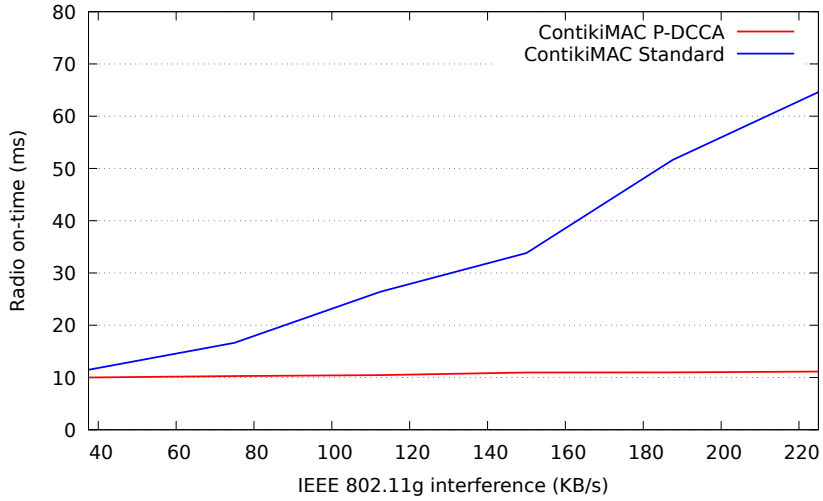
The results are shown in figure 5.11. Figure 5.11a shows that, for both standard and P-DCCA ContikiMAC, there is a negative correlation between interference rate and PRR. This is firstly due to the rate of packet collisions increasing as the interference rate increases. Secondly, due to CCA collisions in ContikiMAC, which stop the transmission sequence. The graph shows that the PRR of ContikiMAC is markedly improved with P-DCCA: under the heaviest interference tested, PRR of standard ContikiMAC falls below 5%, while P-DCCA achieves above 50%. This improvement stems from the simple DCCA interference policy, which ignores non-WSN CCA collisions.

When subjected to the lowest interference level, there is a significant difference between standard and P-DCCA ContikiMAC. This may be the result of other interference in the environment which could not be controlled. In future controlled interference experiments, the range of interference levels will include zero - to be able to evaluate this objectively. These results are still sufficient however to evaluate the performance of ContikiMAC with and without P-DCCA, therefore meeting the objective of this experiment.

Figure 5.11b shows, for standard ContikiMAC, that the radio-on time per packet received increases proportionally with the interference rate. This firstly stems from reduced idle listening caused by false wakeups, and secondly from reduced PRR which affects this measure. By contrast, P-DCCA is much less affected: standard ContikiMAC increases radio-on time to $65ms$ under interference, while P-DCCA increases to only $12ms$. In both cases, $10ms$ is the minimum achieved radio-on time; this reflects the



(a) ContikiMAC Packet Reception Rate under interference with P-DCCA



(b) ContikiMAC Radio-on time per packet received under interference with P-DCCA

Figure 5.11: ContikiMAC with and without P-DCCA under interference

guard time in ContikiMAC when a sender begins transmitting packets, before the receiver is expected to wake up. The improvement yielded by P-DCCA stems firstly from the improvement in PRR, and secondly due to the DCCA response policy, which avoids false wakeups.

Observing standard ContikiMAC in figures 5.11a and 5.11b, PRR is worse affected than radio on-time under low interference rates. This is likely due to the fast sleep optimisation implemented in the ContikiMAC protocol, which mitigates against false wakeups.

5.7.3 Collision policy evaluation

The previous experiment showed that P-DCCA is able to improve the performance of a single wireless link in an environment subject to interference. In this case, only one ContikiMAC node was transmitting. Therefore, the simple response policy improves packet delivery by ignoring collisions. This may not be the case in a deployment with multiple ContikiMAC nodes trying to access the channel - since this collision policy may prevent efficient channel arbitration between WSN nodes. In these cases, channel arbitration may be improved with P-DCCA. For example, upon collision with a WiFi signal, the more optimal approach in this implementation is to persist with the transmission. Conversely, collision with another ContikiMAC packet should result in a random back off to try again. Therefore, the objective of this experiment is to evaluate the simple DCCA collision policies described, in a multiple WSN transmitter environment.

In this experiment, an 802.11g network transmitted 1500 byte packets, at a rate of $37.5KB/s$. This interference rate was chosen following the results of the previous experiment. At this interference rate, measurable packet loss within the WSN network is ensured, however communication is not prevented entirely and therefore the effects of P-DCCA can be measured.

Nodes in the WSN were programmed with CSMA to handle retransmissions, initiating a random backoff after every unsuccessful transmission attempt. On each node, upon an

acknowledgement being received or packet timeout, a new transmission was initiated - attempting maximum throughput, which is then recorded.

In a single-node environment, where all interference is known to have originated outside the network, a trivial solution is to ignore all collisions during packet transmissions. In a multiple-node environment, this is no longer the case since interference may be due to environmental interference or another device in the same network. Consequently, it is expected that DCCA should provide greater benefit. Therefore, as well as standard and P-DCCA ContikiMAC, a third derivative was included to evaluate this: NO-CCA. Here, no CCA checks are conducted either before, or during transmission. In a single transmitter use case, this would be advantageous, since the only source of interference is guaranteed to have originated from outside the network, and an aggressive policy will likely not have a negative impact on the network.

The results are shown in figure 5.12. The results show, firstly, that average throughput per node decreases as the number of transmitters increases. This is to be expected, as there is more channel contention between nodes.

For the case of only one transmitter, NO-CCA achieves higher throughput than P-DCCA. This is due to false positives in P-DCCA - incorrectly detecting 802.15.4 packets and backing off. With multiple transmitters, however, P-DCCA achieves higher throughput compared to both NO-CCA and standard ContikiMAC. This stems from the simple DCCA policy, which avoids collisions only with other WSN transmissions. Also, for more than one transmitter, NO-CCA achieves lower throughput compared to standard ContikiMAC. This is because the NO-CCA collision policy is detrimental to channel arbitration between WSN nodes.

This experiment has evaluated the benefits of P-DCCA in a multiple-transmitter environment. The results have shown that the simple DCCA policy to interference collisions is sufficient to mitigate the effects of interference, while still allowing channel arbitration between WSN nodes.

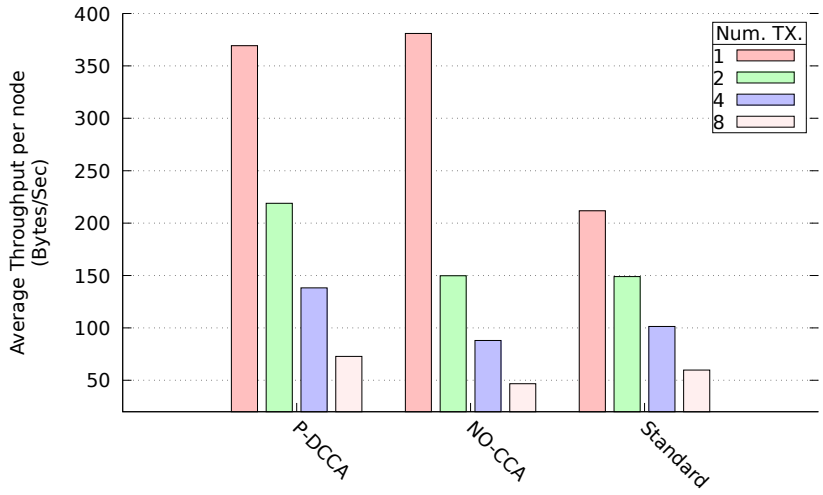


Figure 5.12: ContikiMAC performance with multiple transmitters under WiFi interference

5.7.4 Large scale testbed evaluation

Implementing P-DCCA with simple interferer-detection policies has been shown to reduce the energy costs and improve the packet reception and energy efficiency under interference. Previous experiments have evaluated P-DCCA for small, single-hop networks, with no routing protocol implemented. In a larger network spanning multiple hops, the effect of reduced transmission range of P-DCCA may reduce the number of usable links in a network. Likewise, the inclusion of a routing protocol above the link layer may behave differently with P-DCCA. Therefore, the objective of this experiment is to evaluate the performance of P-DCCA in unison with a routing protocol and multi-hop links.

In order to use a large number of nodes in an automated fashion, the WISEBED [CPC+12] testbed at the University of Lübeck was used. This testbed is based on the 2nd floor of an office building, and consists of 162 nodes arranged in clusters of three. Each node is connected to a host laptop, and the testbed is coordinated over an Ethernet back-end, which allows for node communication and reprogramming. To aid network configuration, all of the nodes have pre-allocated MAC addresses. For this experiment, only the telosB nodes were used.

In order to coordinate communication across the entire network, all nodes were programmed with the RPL routing protocol. RPL was chosen because it supports

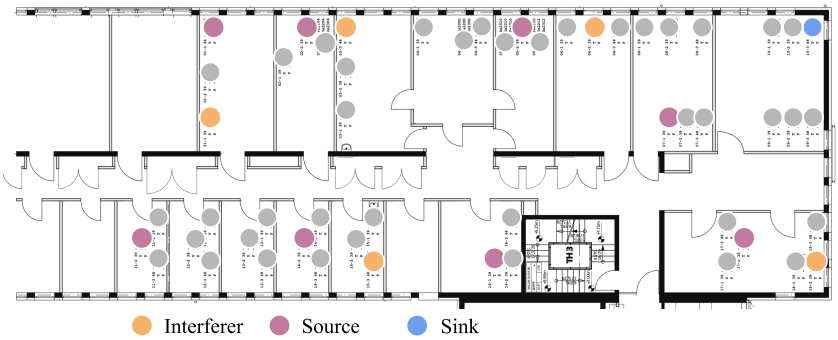


Figure 5.13: WISEBED WSN testbed

the IPv6 protocol stack, and because it is common in real WSN deployments. Also, because the design of RPL is similar to other common routing protocols such as collection tree protocol. RPL establishes an acyclic graph from a sink node to each node in the network. Each node then has a single parent, to forward data toward the sink, and a number of children, for downstream traffic. The IPv6 protocol stack was used to handle packet forwarding and address assignment. The default Contiki CSMA protocol and configuration was used to handle retransmissions This uses an exponentially increasing backoff and a maximum retry limit of three.

In order to generate sufficient traffic within the WSN to measure the effects of interference, eight of the 49 nodes in the network were configured as sources. These generated a 60-byte packet every 30 seconds, sent to a single source node located at one end of the deployment. This network architecture is typical of a data collection WSN deployment. The remaining nodes were part of the network, forwarding packets as required to the sink.

It was necessary to generate repeatable and reliable interference under controlled parameters in the testbed. The WISEBED deployment only supports IEEE 802.15.4-based devices, and therefore other interference - such as WiFi - cannot be generated. Therefore, WiFi interference was simulated by other telosB nodes in the testbed, based on the approach described in JamLab [BVN+11]. These nodes used the test transmission mode of the CC2420, whereby psuedo-random data is transmitted continuously on the same IEEE 802.15.4 channel. The nodes alternate between maximum and minimum transmission power, to simulate packetized IEEE 802.11g traffic. It was desirable to

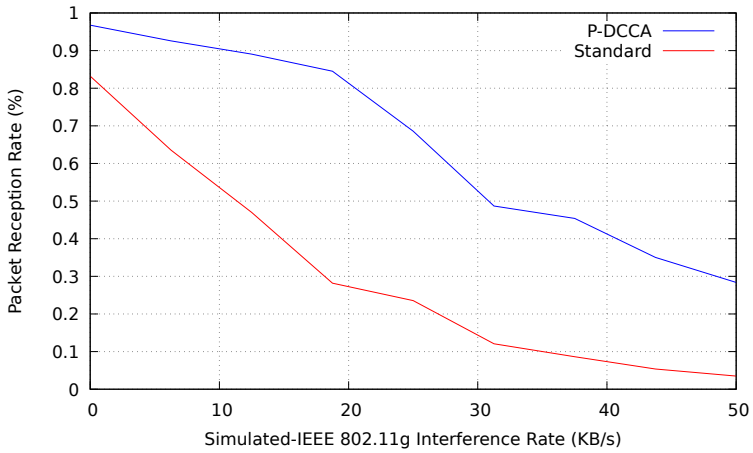


Figure 5.14: ContikiMAC packet reception rate in a large scale testbed, under simulated interference

simulate similar interference conditions to previous experiments, therefore data packets of fixed sized 1500-bytes, transmitted at $54MB/s$, were simulated. The timing between each packet was calculated randomly over a time window that is changed to throttle the degree of interference. Due to technical limitations, this approach does not simulate the IEEE 802.11g MAC protocol. This approach allows for an evaluation of P-DCCA over a multi-hop network in the presence of an interference source. In JamLab [BVN+11], the authors showed that this approach is able to achieve high accuracy of WSN performance compared to the real interferer.

The arrangement of sink, sources, other nodes, and interferer nodes in the testbed is shown in figure 5.13. The sink node was deliberately placed at the periphery of the network, to allow for longer routes across the network. The source nodes were randomly distributed across the network.

The source and sink nodes reported each packet transmission and reception event respectively. Also, all nodes in the network periodically reported the radio-on time from the beginning of the experiment. This is used to calculate the average-radio on time, per packet received at the source. This metric is used since it represents the network-wide energy cost of communicating a packet to the sink. P-DCCA and standard ContikiMAC variants were tested in 15 minute iterations, for each interference level. The experiment was repeated four times.

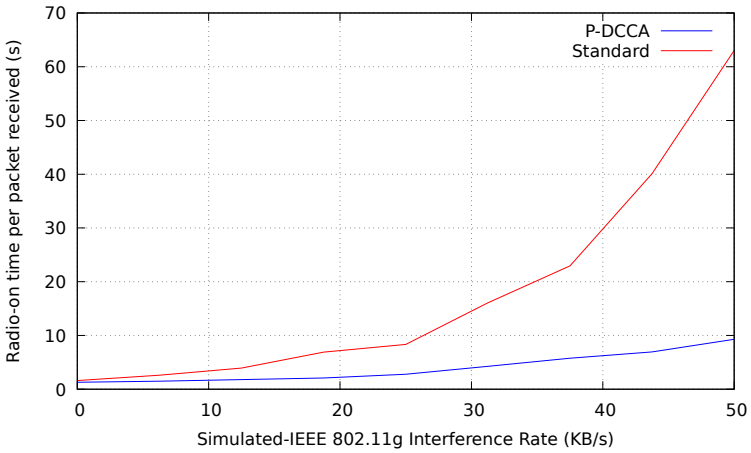


Figure 5.15: ContikiMAC radio-on time per packet received in a large scale testbed, under simulated interference.

The results are shown in figures 5.14 and 5.15. The graphs show that for all simulated interference levels, ContikiMAC with P-DCCA achieves better PRR and radio-on time compared to standard ContikiMAC. A number of other observations can be made from these graphs.

Firstly, from observation, figure 5.14 appears to exhibit a linear relationship between interference rate and PRR, for both ContikiMAC variants. This is not the case with radio-on time in figure 5.15, which does not appear to be linear in either case. This may be attributed to the radio-on time metric being more sensitive to interference in two regards: firstly, due to false wakeups in the MAC protocol; secondly, due to fewer packets being received, thus increasing the per-packet measurement.

Secondly, figure 5.14 shows that even with no simulated interference, some packet loss is still present: PRR of standard and P-DCCA ContikiMAC is 83% and 97% respectively. This is most likely due to other interference sources in the testbed environment. Such interference sources were out of control of the testbed configuration. A longer experiment duration over multiple days may have dampened the effect on the results of this interference. However the results remain sufficient to meet the experiment objectives.

Similarly, the PRR measurements shown in figure 5.14 appear to show noticeable variance. This also may the the result of other interference sources in the testbed environment.

Thirdly, in this experiment, interference is shown to have a greater effect on P-DCCA, compared to previous experiments. PRR is reduced from 97% to 28%, and radio on time is increased from 1290ms to 9296ms, a 71% decrease and 620% increase respectively. This may be due to the cumulative reduction in PRR which is felt across all links on a path - as opposed to the affect on an individual link as measured previously. Also, this experiment simulated interference originating from multiple 802.11 interferers, as opposed to a single, albeit higher power, interferer. This may more greatly affect WSN links.

The energy consumption (figure 5.15) shows similar trends to earlier experiments. The baseline under no interference is measured as 1.3ms. This is lower than the radio-on time measured for a single link (as in figure 5.11b), as this is averaged across all nodes in the network - including nodes that do not participate in packet forwarding.

These results confirm that P-DCCA benefits link quality and energy efficiency, on an individual link level, and also in large multi-hop networks. Importantly, the reduced transmission range stemming from P-DCCA power modulation does not impair packet delivery. This is due to either link availability and quality being relatively unimpaired by this power modulation, or whose affects are negligible in large networks. Consequently, these results show that the drawbacks of employing P-DCCA are far outweighed by the benefits, in this case.

5.7.5 Uncontrolled environment evaluation

In previous experiments, P-DCCA has been shown to improve the link performance and energy efficiency of ContikiMAC. In these cases, the interference environment was controlled by injecting controlled WiFi interference. In a real deployment, the interference may differ from these conditions, and may be the product of many combinations of interference devices and their interactions. For example, this may include Bluetooth and WiFi devices, or other devices emitting interference in the 2.4Ghz frequency range. This cannot be trivially modelled or generated in a controlled interference experiment. Therefore, the objective of this experiment is to evaluate P-DCCA in uncontrolled

interference conditions.

A busy office was chosen as the location for this experiment. Prior measurements observed WiFi interference from nearby laptops and smartphones, as well as other background sources. This location was therefore ideal for evaluating P-DCCA under strenuous uncontrolled interference conditions.

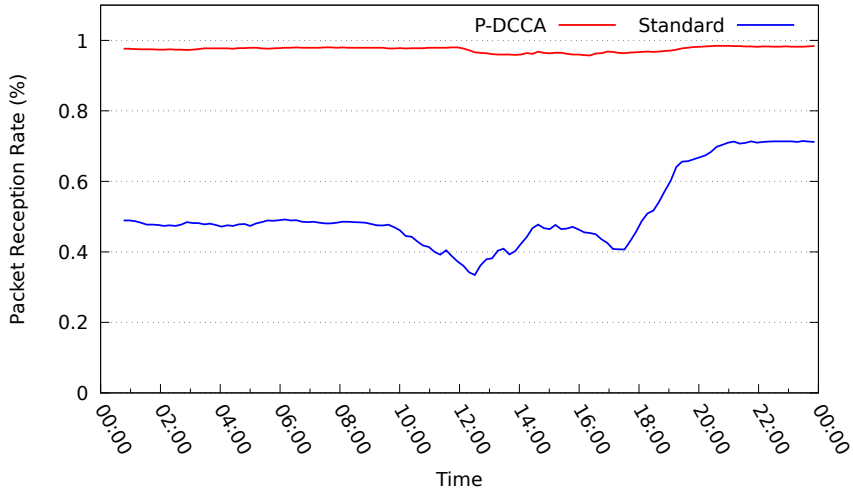
The sender attempted to transmit packets at a fixed rate of four packets/second. This high data rate - in comparison to previous experiments - was chosen to maximise the stress on the network, in order to compare the two ContikiMAC variants. Two variants were compared: standard ContikiMAC and ContikiMAC with P-DCCA, each running on both nodes for five minute intervals.

This experiment only evaluated single-hop link conditions, and therefore only two nodes were used. Likewise, no routing protocol is implemented. In order to capture a range of interference types and intensities, the experiment lasted 24 hours and took place during the work week. Therefore, activity within the office varied as people arrived at work, attended meetings, etc. As previously, PRR and Radio-on time were recorded.

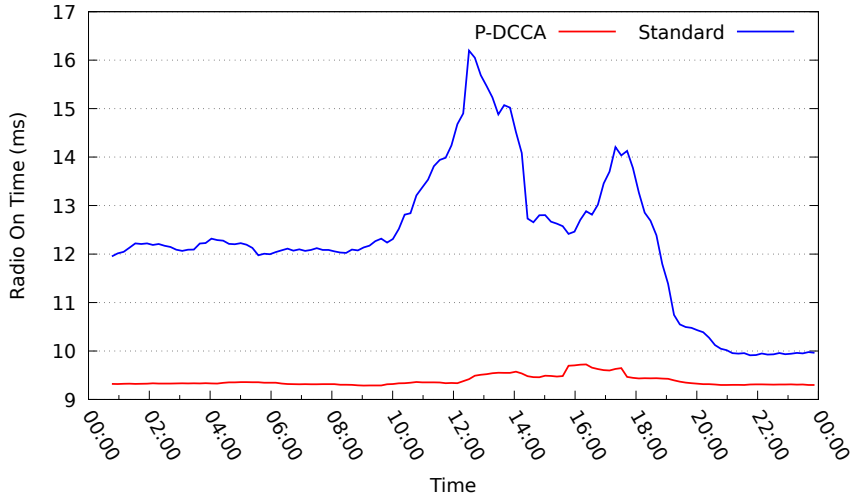
The results are shown in figure 5.16. Both graphs firstly show variation in energy efficiency and PRR through the 24-hour period. This stems from interference from wireless laptops, smartphones, and other $2.4Ghz$ interference. This varies throughout the day as people arrive into work, attend meetings, have lunch etc. The observations from the PRR and radio-on time results coincide in the time domain, as expected. The worst interference seems to occur between 10:00, and 20:00, and spikes in interference appear at 13:00 and 18:00.

The first observation from these graphs is that the variation in PRR and radio on time is noticeably less for the case of P-DCCA, compared to standard ContikiMAC. This is due to implemented DCCA policies not being effected by interference, as per previous experiments.

Secondly, even during quiet office hours, standard ContikiMAC performed worse compared to P-DCCA. PRR was at least 38% higher, and radio-on time at least 6/6% lower with P-DCCA. This shows that during the experiment, there was a minimal level



(a) ContikiMAC Packet Reception Rate in an uncontrolled interference environment



(b) ContikiMAC Radio-on time per packet received in an uncontrolled interference environment

Figure 5.16: Packet Reception Rate and Radio on-time over a 24-hour deployment

of background interference in the office - sufficient to degrade performance of standard ContikiMAC.

Both graphs show asymmetry between the beginning of the experiment and the end. For example, PRR of standard ContikiMAC is 48% at the beginning, and 71% at the end of the experiment. This suggests that the level of background interference was not the same for both days. Reflecting on these results, extending the experiment over a number of days may have provided more insight into the interference patterns in this environment.

Broadly speaking, the results mirror earlier experiments: P-DCCA improves link performance and energy efficiency in ContikiMAC under wireless interference. At peak office hours during the experiment, PRR was improved by upto 180%, and energy efficiency improved by upto 40%.

5.7.6 Discussion

In this section, P-DCCA was implemented in the ContikiMAC WSN MAC protocol, in order to evaluate the prospective improvement DCCA may have on WSN communications under interference. Only simple interferer-detection policies were implemented, whereby all non-WSN traffic is ignored when receiving or transmitting.

More complex policies could seek to optimise coexistence between WSN and other devices further. For example, transmitters could employ a random backoff after any collision, whose duration is dependant on the type of interferer. Likewise, receivers could extend, on demand, the listen duration in ContikiMAC if a packet has been detected but not correctly received - possibly due to packet corruption.

ContikiMAC with, and without, P-DCCA has been tested in controlled, and uncontrolled interference environments. In both cases, P-DCCA was shown to improve link performance - as measured by packet delivery and throughput. The energy efficiency of ContikiMAC was shown to be improved on nodes using P-DCCA, as measured by the radio-on time per packet received. This is due to reduced false wakeups, and consequently less idle listening, as well as the improved packet reception rate.

These results were replicated as well in a large scale testbed experiment, although the improvement in energy efficiency is dampened due to the large number of nodes participating in the network. The quality and availability of links using P-DCCA power variation is unaffected compared to standard, full power transmission. RPL was used to implement the routing protocol, however the semantics, from a link-level perspective, are no different from other such routing protocols - such as collection tree protocol.

5.8 Chapter Summary

In this chapter, a detection solution to mitigate the effects of CTI has been presented, in order to address the problem P.2 raised in section 1.1. Previous detection solutions are known to require higher idle listening cost, reducing energy efficiency in WSN deployments. In this chapter, DCCA was presented as a detection solution which is compatible with existing MAC protocols, requires no special hardware, and does not significantly increase the energy cost.

DCCA is a conceptual extension to the standard 802.15.4 CCA mechanism, which is able to indicate both the presence, and the type, of another signal on the same channel. P-DCCA implements DCCA by varying the output power during transmission, then detecting this during short receiver-checks.

The accuracy of P-DCCA was compared empirically with a similar approach in literature in Section 5.4, and the benefits and drawbacks of each approach discussed in Section 5.4.4. The radio use of P-DCCA was modelled, and shown to achieve greater energy efficiency than current state-of-the-art alternatives to mitigate CTI. P-DCCA was evaluated under controlled and uncontrolled interference conditions, and was shown to improve packet delivery and energy efficiency metrics compared to the standard implementation. In a large-scale multi-hop WSN deployment, P-DCCA was shown to improve both metrics again compared to the standard implementation.

Chapter 6

Conclusion

The conclusions of this thesis are presented in this chapter. Firstly, the premise - which is the effects of CTI acting upon WSN - is revisited from the perspective of the background and related work discussed in Chapter 3. This was described in section 1.1 as packet loss and energy efficiency impairment under interference conditions.

Following this, the problem statement described in section 1.1 - which was derived from these adverse effects - is reviewed. The contributions of this thesis are then reflected upon, with regard to the original problem statement. The scope for future work extending these contributions is then discussed, before some closing remarks which bring the chapter and thesis to a conclusion.

6.1 Thesis Discussion

Due to the advancement of hardware and software design, the ongoing realisation of the IoT, and broader demand, WSNs have become more prominent over the past decade. Deployments have stretched beyond academic fields of interest, to also include industrial, office, and residential applications. Combined with the proliferation of wireless technologies operating in the $2.4Ghz$ ISM band, the issue of CTI is now unavoidable for some WSN designers. Likewise, other technologies which share similarities with WSN, such as home automation, vehicular sensor networks, and the IoT, face the same issues.

In this thesis, the costs of CTI on WSNs were considered twofold: packet delivery and

energy inefficiency. Based on the 802.15.4 PHY protocol, LPL protocols were considered as a specific subclass of WSN MAC protocols - common in applications favouring low energy consumption and data rate. Here, CCAs are used for transmitting and receiving, enabling low duty cycles and long deployment lifetimes. As discussed in Chapter 2, the standard CCA mechanism - Energy Detection - is susceptible to other interference. Previous works, discussed in Chapter 3, have shown this may lead to suboptimal performance.

In chapter 3, previous work relating to WSN and other 802.15.4 network coexistence with interference was discussed. This included experimental and theoretical studies of the effects of interference; models of interference and energy consumption in WSN; and solutions to mitigate the effects of CTI.

This firstly reaffirmed in section 3.2 the adverse effects of interference on WSN. Namely, for IEEE 802.15.4 networks, packet loss and energy efficiency of WSN are impaired under CTI conditions. This is exacerbated for WSN MAC protocols, also further reducing the lifetime of battery-powered networks. The findings in this chapter motivated the remaining work in the rest of the thesis.

Following this, previous simulation and theoretical models of energy consumption for WSN were reviewed in section 3.3. These tools allow the energy consumption of WSN to be estimated before deployment, to allow devices to be adequately provisioned and tuned to meet network lifetime requirements. It was shown that currently link quality can be modelled under interference conditions, and energy consumption under general terms. However, it was shown that there is no intersection of these domains: to model energy consumption of WSN under interference. Consequently, predicting the energy consumption and lifetime of a WSN prior to deployment in interference conditions is not possible with existing state-of-the-art tools. This constitutes problem **P.1**, which was described in section 1.1.

Then, current solutions to mitigating the effects of CTI in 802.15.4 networks and WSN were discussed in chapter 3. Solutions were classed based on their strategy: drawn from avoidance, detection, and resilience approaches. The tradeoffs in each case were considered. Avoidance approaches were shown to offer the greatest potential to mitigate

interference, but require reliable communication between nodes. Detection and resilience approaches are orthogonal, and may be deployed in parallel. It was concluded here that no current state-of-the-art WSN-based detection approaches exist that are sufficiently accurate; do not rely on atypical hardware; and do not significantly increase idle listening. This constitutes problem **P.2**, which was described in section 1.1.

The two primary contributions of this thesis followed from these problems, **P.1** and **P.2**, and are discussed in detail in the remainder of this section.

Problem P.1: Accurate energy consumption estimation of WSN

In Chapter 4, tools were presented which enable WSN designers to estimate the duty cycle of LPL protocols, based on measurements of interference. This chapter began with a discussion of interference measurement techniques available, which have previously been used for link reliability modelling and interference classification. This section concluded that it is sufficient to measure P_C, the channel busy probability.

Following this, a closed form estimation of WSN duty cycle was presented, based on the popular ContikiMAC protocol. The complexity of this approach stemmed from the ContikiMAC state machine, through which numerous paths exist for the receiver to take. This approach is largely inflexible to modifications of how ContikiMAC detects and responds to a busy channel. Therefore, modifying this approach to account for changes to the MAC protocol wakeup or channel sensing mechanism is inherently difficult.

These drawbacks motivated the development of a Monte Carlo solver approach, which takes as input a Lua script representing the MAC protocol wakeup sequence. This approach more closely follows the actual implementation in software of ContikiMAC, and is more accommodating to MAC protocol changes. Both approaches were shown to predict closely the impact of P_C on the duty cycle.

The accuracy of the Monte Carlo solver was then evaluated under controlled interference conditions in a testbed, and uncontrolled conditions in realistic deployments. For typical WSN traffic rates, the solver was shown to be accurate. While the ContikiMAC MAC protocol was used in this chapter as an example, the techniques presented are

applicable to any WSN MAC protocol that relies on CCA for synchronisation.

Problem P.2: Improved detection mechanism in WSN MAC protocols

DCCA was presented in Chapter 5 as a detection solution to mitigate the effects of CTI. DCCA is a conceptual extension to the standard CCA mechanism. As well as indicating the channel as busy/free, DCCA can indicate the source of channel contention. This can then be used to optimise collision response policies, and eliminate false wakeups in the receiving sequence - therefore able to mitigate the main consequences of CTI.

Three methods of implementing DCCA were considered. MD-DCCA was considered the most efficient, although not possible on typical WSN hardware. T-DCCA and P-DCCA were then considered as alternatives not reliant on atypical hardware. The former was based on a similar approach in literature [ZCW+14], while the latter was developed in this thesis. P-DCCA works by varying the output transmission power, then detecting this characteristic feature at the receiver.

An application evaluation then followed, wherein P-DCCA was implemented in ContikiMAC, and evaluated in terms of PRR and radio on-time. For a small, single-hop network, P-DCCA was shown to improve packet delivery performance and energy efficiency, compared to standard ContikiMAC. These results were mirrored in an office environment subject to realistic interference conditions. The same experiment was then repeated on a large, 49-node WSN testbed, where similar results were achieved.

In this chapter, DCCA was shown firstly to be an effective approach to mitigating interference in heterogeneous network environments. This is a novel concept to mitigating channel contention in WSNs: even simple interference response policies were able to significantly improve performance of ContikiMAC under interference. Regardless of the specific implementation, MAC protocols are afforded the ability to arbitrate channel access more efficiently with other devices. How to fully realise this potential remains an open research question.

Secondly, P-DCCA was shown to be an implementation of DCCA that is feasible on current WSN hardware. Compared to T-DCCA, P-DCCA was shown in section 5.4 to

achieve better True Positive- and False Positive- accuracy, despite a shorter sampling duration. This is closer to the standard 802.15.4 CCA implementation, simplifying the process of incorporating DCCA into existing MAC protocols. The tradeoffs of transmission power and range were quantified; these drawbacks were shown to be outweighed by the benefits of DCCA in the application evaluation.

6.2 Future work

Several areas of future work have been identified that extend from the work in this thesis. These range from more advanced CTI-mitigation methods, to potential uses of the transmit power variation mechanism beyond implementing DCCA.

6.2.1 Interferer-response policies

In Chapter 5, the potential benefits of DCCA were demonstrated via simple interferer-response policies. This was sufficient to demonstrate the use case of DCCA in practice, and was able to improve packet delivery and energy efficiency of the WSN in interference environments. More comprehensive policies may, however, yield better performance, or target specific optimisations.

For example, the policies described in Section 5.7 ignore non-WSN devices before transmitting. While this decreased the rate of CCA collisions, and subsequently led to higher packet delivery, packet collisions may increase. Therefore, a more robust approach could be to implement a 1-persistent CSMA policy (similar to 802.11, see figure 2.6) This would ensure a collision with 802.11 is avoided, without an expensive retransmission interval. Similarly, DCCA could further improve the wakeup process by responding differently to packet corruption. If a packet is detected originating from the WSN, the receiver could extend the listen duration even after a corrupted packet is received. This would further improve packet delivery under interference.

It would be beneficial to explore the optimal response policies to various types of interference. The same conclusion has been reached previously by interference classification studies [HRV+13].

6.2.2 Channel prioritisation via P-DCCA

In this thesis, P-DCCA has been used to differentiate WSN traffic from other interference. However, in deployments where CTI is not a concern, P-DCCA may be used to differentiate traffic within a network. This could be used to distinguish source, destination, or types of packet, within a DCCA check.

For example, one application would be to implement distinct traffic priorities, in order to meet design requirements. Downstream traffic may include software updates, network configurations, and node instructions. If this traffic were valued higher than upstream traffic, such as sensor data, it would be advantageous for nodes to distinguish the two, and afford the former greater priority in channel contention. This mechanism would be similar to the 802.11 DCF (see figure 2.6), wherein packet priorities are afforded by the IFS duration. Likewise, traffic priorities could be implemented on a single link, between nodes. For unsuccessful packet transmissions, retransmissions could be afforded higher priority. This would be beneficial in realtime applications, with strict deadlines for packet delivery.

This area of study is only applicable to the transmission power variation used in P-DCCA, and not other DCCA implementations.

6.2.3 Orthogonal channel communication via Transmit power variation

Transmit power variation is used in P-DCCA in order for packets to be differentiated from other interference. This simple modulation method could be extended to encapsulate additional information, in order to provide an orthogonal communication channel. This could be implemented using available amplitude modulation techniques, such as On Off Keying (OOK), and Pulse Width Modulation (PWM). Alternatively, a more specialised modulation technique could be developed.

This could be used firstly to expand the link bandwidth, without requiring a change to the hardware. This could serve a specific purpose, such as forward error correction or security information. Likewise, the receiving mechanism of this channel would not be bound by the same constraints of the 802.15.4 PHY; namely, it is not necessary to

receive and decode the preamble and SFD. This could be leveraged to communicate high priority information, without having to reserve capacity within the 802.15.4 channel. An example of this would be broadcasting TDMA schedules or security keys. In both cases, a new node would be able to join an existing network by overhearing other network traffic.

This area of study would require research to extend the power variation technique, and also a receive mechanism. In both cases, there would be a likely tradeoff between communication bandwidth, and error rate.

6.2.4 Interference mitigation aware energy estimation

The techniques presented in Chapter 4 to estimate energy consumption could be extended, to also model various CTI-mitigation mechanisms. For example, the idle listening of a WSN node could be estimated and compared for various mitigation methods. These could include P-DCCA, T-DCCA, and other approaches from literature [SHL13]. This would similarly be an intersection between energy estimation techniques and interference coexistence studies.

WSN designers would benefit from being able to compare mitigation approaches before deployment. Likewise, it would be possible to optimise parameters for each approach without exhaustive testing. This would be able to account for the respective tradeoffs in each approach to suit the deployment environment. For the case of P-DCCA and T-DCCA, evaluated in Chapter 5, it would be trivial to extend these energy estimation techniques. This would require firstly accounting for the change in CCA duration; for example, in T-DCCA this would be $2.9ms$. Secondly, the CCA function would need to account for the respective False Positive accuracy - as measured in Section 5.4.

6.2.5 Reactive MAC protocol duty cycling

In Chapter 4 it was shown that the idle listening time of a node can be accurately estimated from interference measurements in an environment. This is useful to predict sensor network lifetimes before deployment, allowing for proactive measures to meet

lifetime goals.

To complement this, reactive measures can be used in response to variable interference conditions. To achieve this, the channel can be periodically sampled to measure the channel busy probability. From this, the idle listening can be calculated, either from a closed-form estimation (as in Section 4.3), or from a lookup table. The wakeup frequency can then be adjusted on demand, to meet energy consumption requirements. Consequently, WSNs would be able to ensure lifetime requirements are met, without having to commit to statically configured suboptimal network parameters.

6.3 Concluding Remarks

From smartphones equipped with WiFi to Bluetooth-enabled kettles, advances in wireless communication have delivered innovations to consumer, academic, and research applications. This has led to an abundance of devices communicating in, among others, the $2.4Ghz$ ISM frequency domain. For WSNs, which are dependent on reliable, low power operation, this cost is incurred via packet loss, and energy inefficiency. How to predict in advance and mitigate the affects of CTI are therefore pressing questions amongst WSN research.

In this thesis, it has been shown that the idle listening of a WSN node can be accurately predicted in a known interference environment. This enables network designers to optimise design parameters to meet lifetime goals, without extensive testing or development work prior.

Secondly, DCCA has been developed as a novel extension to the standard 802.15.4 CCA, to cater for more efficient MAC protocol responses to interference. This has been shown to improve link performance, and reduce the energy consumption of affected WSNs. P-DCCA was presented as one possible implementation of DCCA, applicable to current WSN hardware. This work has carved a path for WSN MAC protocols to avoid the one-size-fits-all approach to collision response, in favour of a source-specific approach. In a truly heterogeneous interference environment, this is likely to be a fundamental building block of CTI-resilient WSN.

Bibliography

[AAM11] Junaid Ansari, Tobias Ang, and Petri Mähönen. Wispot: fast and reliable detection of wi-fi networks using ieee 802.15. 4 radios. In *Proceedings of the 9th ACM international symposium on Mobility management and wireless access*, pages 35–44. ACM, 2011. [Cited on page 61]

[ABFS08] Leopoldo Angrisani, Matteo Bertocco, Daniele Fortin, and Alessandro Sona. Experimental study of coexistence issues between ieee 802.11 b and ieee 802.15. 4 wireless networks. *IEEE Transactions on Instrumentation and Measurement*, 57(8):1514–1523, 2008. [Cited on pages 31 and 36]

[ABM+11] Muhammad Mahtab Alam, Olivier Berder, Daniel Menard, Thomas Anger, and Olivier Sentieys. A hybrid model for accurate energy analysis of wsn nodes. *EURASIP Journal on Embedded Systems*, 2011:4, 2011. [Cited on pages 38 and 42]

[AGE+04] Stefano Avallone, S Guadagno, Donato Emma, G Ventre, et al. D-itg distributed internet traffic generator. In *Quantitative Evaluation of Systems, 2004. QEST 2004. Proceedings. First International Conference on the*, pages 316–317. IEEE, 2004. [Cited on pages 110 and 125]

[All09] ZigBee Alliance. Ieee 802.15. 4, zigbee standard. *On http://www. zigbee. org*, 2009. [Cited on pages 16 and 30]

[BKM+12] Nouha Baccour, Anis Koubâa, Luca Mottola, Marco Antonio Zúñiga, Habib Youssef, Carlo Alberto Boano, and Mário Alves. Radio link quality estimation in wireless sensor networks: a survey. *ACM Transactions on Sensor Networks (TOSN)*, 8(4):34, 2012. [Cited on page 49]

[Blu10] SIG Bluetooth. Bluetooth specification, 2010. [Cited on page 23]

[BVN+11] Carlo Alberto Boano, Thiemo Voigt, Claro Noda, Kay Römer, and Marco Zúñiga. Jamlab: Augmenting sensornet testbeds with realistic and controlled interference generation. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 175–186. IEEE, 2011. [Cited on pages 26, 34, 131, and 132]

[BVT+10] Carlo Alberto Boano, Thiemo Voigt, Nicolas Tsiftes, Luca Mottola, Kay Römer, and Marco Antonio Zúñiga. Making sensornet mac protocols robust

against interference. In *Wireless Sensor Networks*, pages 272–288. Springer, 2010. [Cited on pages 21, 34, 35, and 36]

[BYAH06] Michael Buettner, Gary V Yee, Eric Anderson, and Richard Han. X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 307–320. ACM, 2006. [Cited on pages 18 and 34]

[BZRV12] Carlo Alberto Boano, Marco Antonio Zuniga, Kay Römer, and Thiemo Voigt. Jag: Reliable and predictable wireless agreement under external radio interference. In *Real-Time Systems Symposium (RTSS), 2012 IEEE 33rd*, pages 315–326. IEEE, 2012. [Cited on pages 44, 50, and 53]

[BZV+10] Carlo Alberto Boano, Marco Antonio Zúniga, Thiemo Voigt, Andreas Willig, and Kay Romer. The triangle metric: Fast link quality estimation for mobile wireless sensor networks. In *Computer Communications and Networks (ICCCN), 2010 Proceedings of 19th International Conference on*, pages 1–7. IEEE, 2010. [Cited on page 49]

[CPC+12] Geoff Coulson, Barry Porter, Ioannis Chatzigiannakis, Christos Koninis, Stefan Fischer, Dennis Pfisterer, Daniel Bimschas, Torsten Braun, Philipp Hurni, Markus Anwander, et al. Flexible experimentation in wireless sensor networks. *Communications of the ACM*, 55(1):82–90, 2012. [Cited on page 130]

[DDHC+10] Prabal Dutta, Stephen Dawson-Haggerty, Yin Chen, Chieh-Jan Mike Liang, and Andreas Terzis. Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 1–14. ACM, 2010. [Cited on page 19]

[DEA06] Ilker Demirkol, Cem Ersoy, and Fatih Alagöz. Mac protocols for wireless sensor networks: a survey. *Communications Magazine, IEEE*, 44(4):115–121, 2006. [Cited on page 17]

[DGV04] Adam Dunkels, Björn Grönvall, and Thiemo Voigt. Contiki-a lightweight and flexible operating system for tiny networked sensors. In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pages 455–462. IEEE, 2004. [Cited on pages 11 and 40]

[DMLM04] Enrique J Duarte-Melo, Mingyan Liu, and Archan Misra. A modeling framework for computing lifetime and information capacity in wireless sensor networks. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*. Cambridge, UK, March, 2004. [Cited on pages 38 and 42]

[DOTH07] Adam Dunkels, Fredrik Osterlind, Nicolas Tsiftes, and Zhitao He. Software-based on-line energy estimation for sensor nodes. In *Proceedings of the 4th workshop on Embedded networked sensors*, pages 28–32. ACM, 2007. [Cited on page 40]

[dPAP08] Rodolfo de Paz Alberola and Dirk Pesch. Avroraz: extending avrora with an ieee 802.15. 4 compliant radio chip model. In *Proceedings of the 3nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, pages 43–50. ACM, 2008. [Cited on page 39]

[DSVA06] Adam Dunkels, Oliver Schmidt, Thiemo Voigt, and Muneeb Ali. Protothreads: simplifying event-driven programming of memory-constrained embedded systems. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 29–42. Acm, 2006. [Cited on page 11]

[Dun11] Adam Dunkels. The contikimac radio duty cycling protocol. 2011. [Cited on pages 18, 19, 20, and 29]

[DVTMD13] Peter De Valck, Lieven Tytgat, Ingrid Moerman, and Piet Demeester. Coexistence aware clear channel assessment: from theory to practice on an fpga sdr platform. In *Proceedings of the 10th European conference on Wireless Sensor Networks*, pages 165–178. Springer-Verlag, 2013. [Cited on pages 44 and 55]

[EH02] Amre El-Hoiydi. Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 5, pages 3418–3423. IEEE, 2002. [Cited on page 17]

[EHD04] Amre 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*, volume 1, pages 244–251. IEEE, 2004. [Cited on pages 18 and 20]

[GLSJ12] Antonio Gonga, Olaf Landsiedel, Pablo Soldati, and Mikael Johansson. Revisiting multi-channel communication to mitigate interference and link dynamics in wireless sensor networks. In *2012 IEEE 8th International Conference on Distributed Computing in Sensor Systems*, pages 186–193. IEEE, 2012. [Cited on pages 44, 48, and 64]

[HAUV14] Thaier Hayajneh, Ghada Almashaqbeh, Sana Ullah, and Athanasios V Vasilakos. A survey of wireless technologies coexistence in wban: analysis and open research issues. *Wireless Networks*, 20(8):2165–2199, 2014. [Cited on page 30]

[HCCG09] James Hou, Benjamin Chang, Dae-Ki Cho, and Mario Gerla. Minimizing 802.11 interference on zigbee medical sensors. In *Proceedings of the Fourth International Conference on Body Area Networks*, page 5. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009. [Cited on pages 44, 56, and 57]

[HG03] Ivan Howitt and Jose A Gutierrez. Ieee 802.15. 4 low rate-wireless personal area network coexistence issues. In *Wireless Communications and Network-*

ing, 2003. WCNC 2003. 2003 IEEE, volume 3, pages 1481–1486. IEEE, 2003. [Cited on pages 40 and 43]

[HHW09] Jan-Hinrich Hauer, Vlado Handziski, and Adam Wolisz. Experimental study of the impact of wlan interference on ieee 802.15. 4 body area networks. In *Wireless sensor networks*, pages 17–32. Springer, 2009. [Cited on pages 32, 44, and 64]

[HRV+13] Frederik Hermans, Olof Rensfelt, Thiemo Voigt, Edith Ngai, Lars-Åke Norden, and Per Gunningberg. Sonic: classifying interference in 802.15. 4 sensor networks. In *Proceedings of the 12th international conference on Information processing in sensor networks*, pages 55–66. ACM, 2013. [Cited on page 143]

[HWW10] Jan-Hinrich Hauer, Andreas Willig, and Adam Wolisz. Mitigating the effects of rf interference through rssi-based error recovery. In *European Conference on Wireless Sensor Networks*, pages 224–239. Springer, 2010. [Cited on pages 44, 53, and 64]

[HXS+13] Pei Huang, Li Xiao, Sima Soltani, Matt W Mutka, and Ning Xi. The evolution of mac protocols in wireless sensor networks: A survey. *Communications Surveys & Tutorials, IEEE*, 15(1):101–120, 2013. [Cited on page 17]

[HXZZ10] Jun Huang, Guoliang Xing, Gang Zhou, and Ruogu Zhou. Beyond co-existence: Exploiting wifi white space for zigbee performance assurance. In *Network Protocols (ICNP), 2010 18th IEEE International Conference on*, pages 305–314. IEEE, 2010. [Cited on pages 44 and 52]

[IEE07] IEEE. Part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans). *IEEE Std. 802.15.4*, 2007. [Cited on page 49]

[Ins06] Texas Instruments. Cc2420: 2.4 ghz ieee 802.15. 4/zigbee-ready rf transceiver. *Available at Available at http://www. ti. com/lit/gpn/cc2420*, page 53, 2006. [Cited on pages 12, 14, 15, 32, and 93]

[Ins09] Texas Instruments. Msp430 f1611 datasheet, 2009. [Cited on page 12]

[isa] Isa 100. [Cited on page 30]

[IWL11] Venkatraman Iyer, Matthias Woehrle, and Koen Langendoen. Chryssoa multi-channel approach to mitigate external interference. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on*, pages 449–457. IEEE, 2011. [Cited on pages 44, 45, 46, 47, 48, 49, 50, and 64]

[JB07] Kyle Jamieson and Hari Balakrishnan. Ppr: Partial packet recovery for wireless networks. In *ACM SIGCOMM Computer Communication Review*, volume 37, pages 409–420. ACM, 2007. [Cited on pages 51, 53, and 64]

[KPC+07] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steven Glaser, and Martin Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *2007 6th International Symposium on Information Processing in Sensor Networks*, pages 254–263. IEEE, 2007. [Cited on page 2]

[KT13] Vikash Kumar and Jawahar Thakur. Multi-channel mac protocols in wireless sensor networks: A review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(10):504509, Oct 2013. [Cited on pages 48 and 64]

[LLSH16] Feng Li, Jun Luo, Gaotao Shi, and Ying He. Art: Adaptive frequency-temporal co-existing of zigbee and wifi. *IEEE transactions on Mobile Computing*, PP(99), 2016. [Cited on pages 44 and 49]

[LMMR07] Andreas Lachenmann, Pedro José Marrón, Daniel Minder, and Kurt Rothermel. Meeting lifetime goals with energy levels. In *Proceedings of the 5th international conference on Embedded networked sensor systems*, pages 131–144. ACM, 2007. [Cited on page 40]

[LPLT10] Chieh-Jan Mike Liang, Nissanka Bodhi Priyantha, Jie Liu, and Andreas Terzis. Surviving wi-fi interference in low power zigbee networks. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 309–322. ACM, 2010. [Cited on pages 33, 36, 51, 57, and 64]

[MCP+02] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97. ACM, 2002. [Cited on page 2]

[MELT08] Răzvan Musăloiu-E, Chieh-Jan Mike Liang, and Andreas Terzis. Koala: Ultra-low power data retrieval in wireless sensor networks. In *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on*, pages 421–432. IEEE, 2008. [Cited on page 34]

[MET08] Razvan Musaloiu-E and Andreas Terzis. Minimising the effect of wifi interference in 802.15. 4 wireless sensor networks. *International Journal of Sensor Networks*, 3(1):43–54, 2008. [Cited on pages 44, 45, 46, 47, and 64]

[MGC16] Mobashir Mohammad, XiangFa Guo, and Mun Choon Chan. Oppcast: Exploiting spatial and channel diversity for robust data collection in urban environments. In *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 1–12. IEEE, 2016. [Cited on pages 48 and 64]

[MHK07] David Moss, Jonathan Hui, and Kevin Klues. Low power listening. *TinyOS Core Working Group, TEP*, 105, 2007. [Cited on pages 29 and 34]

[ML] David Moss and Philip Levis. Box-macs: Exploiting physical and link layer boundaries in low-power networking. [Cited on pages 19 and 60]

[MN14] Antonio Moschitta and Igor Neri. Power consumption assessment in wireless sensor networks. *ICT-Energy-Concepts Towards Zero-Power Information and Communication Technology*, 2014. [Cited on page 39]

[PHC04] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107. ACM, 2004. [Cited on page 18]

[PTH+08] Sofie Pollin, Ian Tan, Bill Hodge, Carl Chun, and Ahmad Bahai. Harmful coexistence between 802.15. 4 and 802.11: A measurement-based study. In *Cognitive Radio Oriented Wireless Networks and Communications, 2008. CrownCom 2008. 3rd International Conference on*, pages 1–6. IEEE, 2008. [Cited on pages 33 and 36]

[PWMR07] Marina Petrova, Lili Wu, Petri Mahonen, and Janne Riihijarvi. Interference measurements on performance degradation between colocated ieee 802.11 g/n and ieee 802.15. 4 networks. In *Networking, 2007. ICN'07. Sixth International Conference on*, pages 93–93. IEEE, 2007. [Cited on pages 31 and 36]

[SCPK05] Soo Young Shin, Sunghyun Choi, Hong Seong Park, and Wook Hyun Kwon. Lecture notes in computer science: packet error rate analysis of ieee 802.15. 4 under ieee 802.11 b interference. In *International Conference on Wired/Wireless Internet Communications*, pages 279–288. Springer, 2005. [Cited on pages 117 and 118]

[SGAP00] Katayoun Sohrabi, Jay Gao, Vishal Ailawadhi, and Gregory J Pottie. Protocols for self-organization of a wireless sensor network. *IEEE personal communications*, 7(5):16–27, 2000. [Cited on page 17]

[SGJ08] Yanjun Sun, Omer Gurewitz, and David B Johnson. Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 1–14. ACM, 2008. [Cited on pages 19 and 35]

[She04] Tmote Sky Data Sheet. Moteiv, san francisco, ca, 2006, 2004. [Cited on pages 12 and 102]

[SHL11a] Mo Sha, Gregory Hackmann, and Chenyang Lu. Arch: Practical channel hopping for reliable home-area sensor networks. In *2011 17th IEEE Real-Time and Embedded Technology and Applications Symposium*, pages 305–315. IEEE, 2011. [Cited on pages 44, 45, 46, 47, and 64]

[SHL11b] Mo Sha, Gregory Hackmann, and Chenyang Lu. Multi-channel reliability and spectrum usage in real homes: Empirical studies for home-area sensor networks. In *Proceedings of the Nineteenth International Workshop on Quality of Service*, page 39. IEEE Press, 2011. [Cited on pages 32, 44, 47, and 64]

[SHL13] Mo Sha, Gregory Hackmann, and Chenyang Lu. Energy-efficient low power listening for wireless sensor networks in noisy environments. In *Proceedings of the 12th international conference on Information processing in sensor networks*, pages 277–288. ACM, 2013. [Cited on pages 44, 58, 59, 65, and 145]

[SPCK07] Soo Young Shin, Hong Seong Park, Sunghyun Choi, and Wook Hyun Kwon. Packet error rate analysis of zigbee under wlan and bluetooth interferences. *IEEE Transactions on Wireless Communications*, 6(8):2825–2830, 2007. [Cited on pages 41 and 43]

[SPK07a] Soo Young Shin, Hong Seong Park, and Wook Hyun Kwon. Mutual interference analysis of ieee 802.15. 4 and ieee 802.11 b. *Computer Networks*, 51(12):3338–3353, 2007. [Cited on pages 41 and 43]

[SPK07b] Soo Young Shin, Hong Seong Park, and Wook Hyun Kwon. Packet error rate analysis of ieee 802.15. 4 under saturated ieee 802.11 b network interference. *IEICE transactions on communications*, 90(10):2961–2963, 2007. [Cited on pages 41 and 43]

[TAJC] Gilles Thonet, Patrick Allard-Jacquin, and Pierre Colle. Zigbee-wifi coexistence. [Cited on page 30]

[TLP05] Ben L Titzer, Daniel K Lee, and Jens Palsberg. Avrora: Scalable sensor network simulation with precise timing. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 67. IEEE Press, 2005. [Cited on page 39]

[TQD+05] Ajay Tirumala, Feng Qin, Jon Dugan, Jim Ferguson, and Kevin Gibbs. Iperf: The tcp/udp bandwidth measurement tool. *htt p://dast. nlanr. net/Projects*, 2005. [Cited on pages 82 and 86]

[TWD+13] Yong Tang, Zhipeng Wang, Tianyu Du, Dimitrios Makrakis, and Hussein T Mouftah. Study of clear channel assessment mechanism for zigbee packet transmission under wi-fi interference. In *2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*, pages 765–768. IEEE, 2013. [Cited on pages 44 and 58]

[TWMM13] Yong Tang, Zhipeng Wang, Dimitrios Makrakis, and Hussein T Mouftah. Interference aware adaptive clear channel assessment for improving zigbee packet transmission under wi-fi interference. In *2013 IEEE International Conference on Sensing, Communications and Networking (SECON)*, pages 336–343. IEEE, 2013. [Cited on pages 44, 58, 59, and 65]

[TYP+12] Lieven Tytgat, Opher Yaron, Sofie Pollin, Ingrid Moerman, and Piet Demeester. Avoiding collisions between ieee 802.11 and ieee 802.15. 4 through coexistence aware clear channel assessment. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):1–15, 2012. [Cited on pages 44 and 55]

[VÖ08] Thiemo Voigt and Fredrik Österlind. Coredac: Collision-free command-response data collection. In *Emerging Technologies and Factory Automation,*

2008. ETFA 2008. IEEE International Conference on, pages 967–973. IEEE, 2008. [Cited on page 34]

[WALR+06] Geoffrey Werner-Allen, Konrad Lorincz, Mario Ruiz, Omar Marcillo, Jeff Johnson, Jonathan Lees, and Matt Welsh. Deploying a wireless sensor network on an active volcano. *IEEE internet computing*, 10(2):18–25, 2006. [Cited on page 2]

[WHY06] Qin Wang, Mark Hempstead, and Woodward Yang. A realistic power consumption model for wireless sensor network devices. In *2006 3rd annual IEEE communications society on sensor and ad hoc communications and networks*, volume 1, pages 286–295. IEEE, 2006. [Cited on page 38]

[wir] Wireless hart technology. [Cited on pages 16 and 30]

[WWZ+11] Yufei Wang, Qixin Wang, Zheng Zeng, Guanbo Zheng, and Rong Zheng. Wicop: Engineering wifi temporal white-spaces for safe operations of wireless body area networks in medical applications. In *Real-Time Systems Symposium (RTSS), 2011 IEEE 32nd*, pages 170–179. IEEE, 2011. [Cited on pages 44, 56, and 57]

[XSL+11] Ruitao Xu, Gaotao Shi, Jun Luo, Zenghua Zhao, and Yantai Shu. Muzi: Multi-channel zigbee networks for avoiding wifi interference. In *Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing*, pages 323–329. IEEE, 2011. [Cited on pages 44, 45, and 46]

[YLN10] Wei Yuan, Jean-Paul MG Linnartz, and Ignas GMM Niemegeers. Adaptive cca for ieee 802.15. 4 wireless sensor networks to mitigate interference. In *2010 IEEE Wireless Communication and Networking Conference*, pages 1–5. IEEE, 2010. [Cited on pages 44 and 58]

[YWLN13] Wei Yuan, Xiangyu Wang, Jean-Paul MG Linnartz, and Ignas GMM Niemegeers. Coexistence performance of ieee 802.15. 4 wireless sensor networks under ieee 802.11 b/g interference. *Wireless Personal Communications*, 68(2):281–302, 2013. [Cited on page 41]

[YXG11] Dong Yang, Youzhi Xu, and Mikael Gidlund. Wireless coexistence between ieee 802.11-and ieee 802.15. 4-based networks: A survey. *International Journal of Distributed Sensor Networks*, 2011, 2011. [Cited on pages 29 and 30]

[ZCW+14] Xiaolong Zheng, Zhichao Cao, Jiliang Wang, Yuan He, and Yunhao Liu. Zisense: towards interference resilient duty cycling in wireless sensor networks. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pages 119–133. ACM, 2014. [Cited on pages 22, 26, 39, 42, 43, 44, 58, 60, 61, 62, 65, 91, 95, 96, 97, 111, 112, 113, and 142]

[ZS13] Xinyu Zhang and Kang G Shin. Cooperative carrier signaling: harmonizing coexisting wpan and wlan devices. *IEEE/ACM Transactions on Networking*, 21(2):426–439, 2013. [Cited on pages 44 and 57]

[ZXX+10] Ruogu Zhou, Yongping Xiong, Guoliang Xing, Limin Sun, and Jian Ma. Zifi: wireless lan discovery via zigbee interference signatures. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 49–60. ACM, 2010. [Cited on page 61]