Universidad Politécnica de Madrid
Federal University of Campina Grande
Doctorado en Ingeniería de Sistemas y Servicios para la Sociedad de la Información
Doutorado em Engenharia Elétrica
Redes, Sistemas, Servicios y Tecnologías de Telecomunicación

Contributions to Industrial Wireless Sensor Networks

Diego Véras de Queiroz

Campina Grande, Brazil
10/2020

Diego Véras de Queiroz

Contributions to Industrial Wireless Sensor Networks

Ph.D. thesis submitted to the Coordination of the Postgraduate Program in Electrical Engineering at the Federal University of Campina Grande in a cotutela agreement with the Universidad Politécnica de Madrid as a necessary requirement for obtaining a Doctor of Science degree in the field of Electrical Engineering / Ingeniería de Sistemas y Servicios para la Sociedad de la Información.

Supervisors:

Marcelo Sampaio de Alencar (UFCG) / Cesar Benavente-Peces (UPM)

Cosupervisor:

Iguatemi Eduardo da Fonseca (UFPB)

UFCG – Federal University of Campina Grande
UPM – Universidad Politécnica de Madrid
UFPB – Federal University of Paraiba

Campina Grande, Brazil

10/2020

Q3c Queiroz, Diego Véras de.
Contribuição para redes de sensores sem fio industriais / Diego Véras de Queiroz. - Campina Grande, 2020.
122f. : il. Color.

Tese (Doutorado em Engenharia Elétrica) - Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, 2020.
"Orientação: Prof. Dr. Marcelo Sampaio de Alencar, Prof. Dr. César Benavente-Peces, Prof. Dr. Iguatemi Eduardo da Fonseca".
Referências.

1. Arquitetura de Redes. 2. Unidades de Processamento. 3. Programação de Computadores. 4. Interconexão de Redes. I. Alencar, Marcelo Sampaio. II. Benavente-Peces, César. III. Fonseca, Iguatemi Eduardo da. IV. Título.

CDU 004.72(043)

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECÁRIA ITAPUANA SOARES DIAS CRB-15/93

Acknowledgements

- I would like to thank and dedicate this thesis to my wife Elane Meireles, which is my big star Sirius A, and my little star Sirius B named Lorena, love of my life;
- To my parents Antonio de Queiroz and Creusa Véras;
- To my in-laws Luiz Carlos and Iclea de Almeida, and brother-in-law Danilo;
- To my supervisors Prof. Marcelo Sampaio de Alencar, Prof. Cesar Benavente-Peces and Prof. Iguatemi Eduardo da Fonseca for all the help with this project;
- To IECOM, to the Radio Engineering Group (GIRA) at UPM, and the Erasmus Mundus Smart^2 (EM-SMART^2);
- To Mrs. Dolores Ajates for the assistance given during my stay at UPM;
- To Professor Ana Belén Garcia Hernando of the UPM Sub-direction for her efforts in the cotutela agreement;
- To Professor Michel François Fossy for the guidelines in the agreement of the UFCG part;
- To all the staff members of UFCG department, especially to Mrs. Ângela de Lourdes Ribeiro Matias, for the guidelines in the course rules;
- To the members of the Superintendence of Information Technology of UFPB, especially Hermes Pessoa Filho and Janio Carlos Mesquita Vieira, Frederico Augusto Monteiro Saraiva and Raniere Fernandes de Medeiros;
- To everyone who contributed to this work, especially Prof. Ruan Delgado Gomes (IFPB);
- To other friends and family for their support.

Science never solves a problem without creating ten more.

George Bernard Shaw (1856-1950)

Contribuições para redes de sensores sem fio industriais

Autor: Diego Véras de Queiroz

Orientadores: Marcelo Sampaio de Alencar, Ph.D., Cesar Benavente-Peces, Ph.D.

Co-orientador: Iguatemi Eduardo da Fonseca, Ph.D.

Resumo

Avanços recentes nas redes de sensores sem fio (WSN), especialmente em ambientes industriais (IWSN), trouxeram melhorias importantes para implantação de uma rede de sensores nesses ambientes. No entanto, há um alto nível de interferência, ruído, sombreamento e desvanecimento por múltiplos percursos na indústria, causados por máquinas, objetos metálicos e obstruções, que podem afetar a qualidade do enlace. Alguns mecanismos foram propostos para mitigar os efeitos negativos, como salto de frequência e lista negra de canais. No entanto, ainda existem questões em aberto, como um método apropriado de gerenciar ambos os mecanismos para evitar canais similares ou adjacentes em transmissões simultâneas. O uso de um estimador de qualidade de enlace (LQE), com uma atribuição adequada de canais, ajuda a aumentar o desempenho da rede. Além do uso de estimadores, esta tese propõe duas abordagens que separam os canais de acordo com o perfil temporário de cada um e os insere em listas. O objetivo é lidar melhor com os efeitos negativos do ambiente no meio de transmissão. A primeira usa lista dupla de canais (adequados e inadequados), e a segunda utiliza uma lista tripla de canais (adequados, inadequados e cinza). Em ambas foi levado em consideração o gerenciamento do deslocamento do canal (*channel offset*), de maneira a evitar colisões e diminuir a interferência interna em transmissões simultâneas. A lista de canais inadequados apresenta canais de baixa qualidade, a lista cinza possui canais com qualidade incerta, e a lista de canais adequados possui canais com boa qualidade. Um método de lógica difusa foi utilizado na abordagem com lista tripla para classificar os canais na lista mais adequada. As duas propostas foram comparadas por meio de simulação utilizando o método de salto em frequência baseado no protocolo TSCH do padrão IEEE 802.15.4e para WSN. No estudo, foram analisadas redes nas topologias em estrela e em árvore, com e sem estimadores de qualidade, com e sem colisões, utilizando um modelo de canal realista para IWSN. Os resultados dos experimentos mostraram que na topologia em árvore as abordagens com lista dupla e tripla apresentaram um melhor desempenho do que a abordagem sem estimadores de qualidade, em termos de taxa de entrega de pacotes e determinismo, e que a lista tripla superou a lista dupla em redes mais dinâmicas. Na topologia em estrela, quanto maior o tamanho da lista de canais inadequados, melhor o desempenho da rede, pois os dispositivos utilizam apenas os melhores canais.

Palavras-chave: Redes de sensores sem fio industriais; salto em frequência; deslocamento de canal; lista de canais; lógica difusa.

Contribución a las redes inalámbricas de sensores en entornos industriales

Autor: Diego Véras de Queiroz
Directores: Marcelo Sampaio de Alencar, Ph.D., Cesar Benavente-Peces, Ph.D.
Cosupervisor: Iguatemi Eduardo da Fonseca, Ph.D.

Resumen

Los avances recientes en redes de sensores inalámbricos (WSN), especialmente en entornos industriales (IWSN), han traído mejoras importantes para el despliegue de redes inalámbrica de sensores en estos entornos. Sin embargo, en estos entornos existe un alto nivel de interferencia, ruido, zonas de sombra y desvanecimiento, propagación multitrayecto en el entorno industrial, causados por reflexiones en maquinaria, objetos metálicos y elementos propios de la construcción, que afectan a la calidad del enlace. Se han propuesto algunos mecanismos para mitigar los efectos negativos de dichas perturbaciones, como el salto de frecuencia y la lista negra de canales. Sin embargo, todavía hay preguntas y líneas de investigación abiertas, como el desarrollo de un método apropiado para gestionar simultáneamente ambos mecanismos para evitar transmisiones simultáneas canales similares o adyacentes susceptibles de interferirse. El uso de un estimador de calidad de enlace (LQE) con asignación del canal más adecuado ayuda a mejorar la calidad de la red. Además del uso de estimadores, esta tesis propone dos enfoques que separan los canales de acuerdo con el perfil temporal de cada uno y los insertan en listas ordenadas. El objetivo es tratar de paliar los efectos negativos producidos por el entorno que afectan al medio de transmisión. El primero usa una lista de canales dobles (blanco y negro, o más bien, adecuado e inadecuado), y el segundo usa una lista triple de canales (blanco, negro y gris, o más bien, adecuado, inadecuado y gris). En ambos, se tuvo en cuenta la gestión de la amplitud del salto de canal (*channel offset*), para evitar colisiones y de este modo reducir la interferencia intra-red en transmisiones simultáneas. La lista de canales inapropiados ordena canales de baja calidad, la lista gris tiene canales con calidad incierta y la lista de canales adecuados, es decir, que lista tiene canales con buena calidad. Para la clasificación de los canales en la lista más apropiada se ha empleado un método basado en lógica difusa en la solución propuesta basada en el uso de la triple lista. Con el fin de su validación, las dos propuestas se compararon mediante simulación utilizando el método de salto de frecuencia basado en el protocolo TSCH del estándar IEEE 802.15.4e para WSN. En el estudio, se analizaron redes en topologías en estrella y árbol, con y sin estimadores de calidad, con y sin colisiones, utilizando un modelo de canal realista para IWSN. Los resultados de los experimentos mostraron que en la topología en árbol, las soluciones propuestas basada en lista doble y triple presentaron un mejor comportamiento que el enfoque sin estimadores de calidad, en términos de tasa de entrega de paquetes y determinismo, y que la solución basada en la lista triple superó a la basada en la lista doble en redes más dinámicas. En el caso de la topología en estrella, cuanto mayor sea la lista de canales inapropiados, mejor será el rendimiento de la red, ya que los dispositivos usan solo los mejores canales.

Palabras clave: Redes inalámbricas de sensores; entornos industriales; salto de frecuencia; cambio de canal; lista de canales; lógica difusa

Contributions to Industrial Wireless Sensor Networks

Author: Diego Véras de Queiroz
Supervisors: Marcelo Sampaio de Alencar, Ph.D., Cesar Benavente-Peces, Ph.D.
Cosupervisor: Iguatemi Eduardo da Fonseca, Ph.D.

Abstract

Recent advances in wireless sensor networks (WSN), especially in industrial environments (IWSN), have brought important improvements for the deployment of a sensor network in such environments. However, there is a high level of interference, noise, shadowing, and multipath fading in industry, caused by machinery, metallic objects and obstructions, which can affect the quality of the link. Some mechanisms have been proposed to mitigate the negative effects, such as frequency hopping and channel blacklisting. However, there are still open issues, such as an appropriate method of managing both mechanisms to avoid similar or adjacent channels in simultaneous transmissions. The use of a link quality estimator (LQE), with adequate channel assignment, helps to increase network performance. In addition to the use of estimators, this thesis proposes two approaches that separate the channels according to the temporary profile of each one and insert them into lists. The aim is to better deal with the negative effects of the environment on the transmission medium. The first uses a double channel list (alowlist and denylist), and the second uses a triple list of channels (denylist, greylist, and alowlist). In both, the management of channel offset was considered to avoid collisions and reduce internal interference in simultaneous transmissions. The denylist has low quality channels, the greylist has channels with uncertain quality, and the alowlist has good quality channels. A fuzzy logic method was used in the triple list approach to classify the channels in the most appropriate list. The two proposals were compared by means of simulation using the frequency hopping method based on the TSCH protocol of the IEEE 802.15.4e standard for WSN. In the study, networks in star and tree topologies were analyzed, with and without link quality estimators, with and without collisions, using a realistic channel model for IWSN. The results of the experiments showed that in the tree topology the approaches with double and triple list presented better performances than the approach without link quality estimators, in terms of package delivery rate and determinism, and that the triple list surpassed the double list in more dynamic networks. In star topology, the larger the size of the denylist, the better the network performance, as the devices use only the best channels.

Keywords: Industrial wireless sensor networks; channel hopping; channel offset; channel listing; fuzzy logic.

List of Figures

List of Tables

List of abbreviations and acronyms

6LoWPAN – IPv6 over Low power Wireless Personal Area Networks

AB-TSCH – Adaptive-Blacklist TSCH

ACK – Acknowledgement Packet

AM – Active Message Protocol

AMCA – Asynchronous Multi-Channel Adaptation

BI – Beacon Interval

BL – Channel Denylist

BLINK – Frequency Identification Blink

BSD – Berkeley Software Distribution

CAP – Contention Access Period

CFP – Contention-Free Period

CH – Cluster Head

CoAP – Constrained Application Protocol

CSMA/CA – Carrier Sense Multiple Access with Collision Avoidance

DSME – Deterministic and Synchronous Multi-channel Extension

EB – Enhanced Beacons

EM-MAC - Energy-Efficient MAC Protocol

FIFO – First In, First Out

FIT/ IoT Lab – Large scale IoT testbed

GTS – Guaranteed Time Slots

IEEE – Institute of Electrical and Electronics Engineers

IETF – Internet Engineering Task Force

IIoT – Industrial Internet of Things

IoT – Internet of Things

IPv6 – Internet Protocol version 6

IWSN – Industrial Wireless Sensor Networks

LLDN – Low Latency Deterministic Network

LLN - Low Power and Lossy Networks

LOS – Line-Of-Sight

LQE – Link Quality Estimator

LQI – Link Quality Indicator

LR-WPAN – Low-rate, Wireless Personal Area Network

MAC - Media Access Control layer

MCU – Microcontroller

NDLNC4 – No denylists, no collisions, four cluster heads

NS2/NS3 – Network Simulator, versions 2 and 3

OMNeT++ – Objective Modular Network Testbed

OS – Operating System

PDR – Packet Delivery Rate

QoS – Quality of Service

RMS – Root-Mean-Square

RNP – Required Number of Packet Transmissions

RPL – IPv6 Routing Protocol for LLNs

RSSI – Received Signal Strength Indication

TDMA – Time Division Multiple Access

TG4e – Task Group 4e

TKN15.4 – Protocol Stack of IEEE 802.15.4 in TinyOS

TMCP – Tree-Based Multi-Channel Protocol

TOSSIM – TinyOS Simulator

TSCH – Time Slotted Channel Hopping

WL – Channel Allowlist

WSN – Wireless Sensor Networks

YDLYC4 – With denylists, with collisions, four cluster heads

YTLNC4 – With triple list, no collisions, four cluster heads

YTLYC4 – With triple list, with collisions, four cluster heads

List of Symbols

n – Path loss exponent

d_0 – Reference distance

$L(d_0)$ – Path loss at reference distance

X_σ – Standard deviation of shadowing

K – Rice factor

K_σ – Standard deviation of the Rice factor

ASN – Number of timeslots of the network since the beginning

FHS_{length} – Number of available channels

N_{ts} – Number of timeslots allocated to coordinators

SF – Slotframe duration in milliseconds

T_{ts} – Timeslot duration in milliseconds

D_{end} – Number of end nodes connected to each cluster head

N_{ch} – Number of cluster heads

Contents

1 Introduction

Traditionally, industrial automation systems are built using wired communications [Queiroz et al. 2017], and these systems present little flexibility, aside from high installation and maintenance costs. To deploy a wired sensor network, a network infrastructure needs to be implemented, which can lead to high costs for adapting the industry and transposing physical barriers for installing the communication cables. In addition, if it is required to change the network topology due to the position of sensor nodes, the network infrastructure needs to be adapted, which can also represent a high cost.

An alternative for building flexible, low-cost monitoring and control systems with easy installation and maintenance, is the use of wireless sensor networks (WSN). Some of the main concerns when designing a WSN are energy efficiency, reliability, and punctuality in communications, and the last two are critical issues in industrial and healthcare applications [De Guglielmo, Brienza and Anastasi 2016].

Recent advances in the research area of WSN, especially in industrial environments (IWSN), have brought important improvements in the effectiveness of deploying a sensor network in such environments. However, an open challenge is the high interference and noise caused by machines, several metallic objects, and obstructions, as depicted in Figure 1, which can affect the link quality.

The use of IWSN is subject to typical problems of wireless communications, such as noise, shadowing, multipath fading, and interference. In addition, the wireless channel in many industrial environments is non-stationary in the long term, which can cause abrupt changes in the characteristics of the channel over time. The channels used for data transmission may temporarily present low quality or even make communication between sensors impossible, given the mentioned problems.

Due to the multipath profile in reflective industrial environments, the coherence bandwidth can be very small, so the characteristics of the communication channel can be

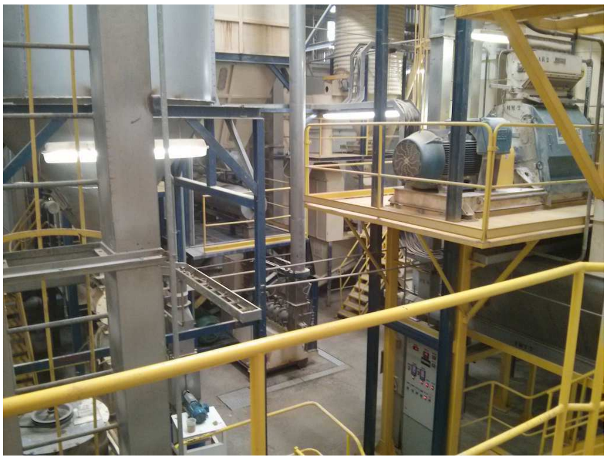

Figure 1: Industrial environment.

very different, even in adjacent channels. Other important aspect is the spatial variation in the quality of the channels, in which a channel can present good quality for some nodes and very low quality for other nodes, even if they are close to each other. These problems make the development of a wireless solution even more challenging.

Many industrial monitoring systems need to process heterogeneous signals that change quickly, and in this type of application, the network operation must occur in a deterministic way, since the monitored information needs to be delivered to the control systems periodically, following the delivery times of each package. In addition, the sensor nodes have resource restrictions, presenting low processing capability and battery limitations.

Another problem that may affect the network performance is the link asymmetry, which is the difference between the quality of the wireless channel in both directions. In WSNs, most packets are generally transmitted in one direction, from the end node to the sink node. However, many protocols use packet recognition (ACK packets) or control packets, and in such cases, it is important to ensure that the link has good quality in both sides.

Despite the aforementioned disadvantages of the wireless media, the most viable alternative for building monitoring and control systems in industrial environments is the use of wireless networks, which are more flexible, present lower cost, they can be self-organized, easy to install and maintain, and have local processing capability.

Some mechanisms were proposed to mitigate the negative effects of the environment on the wireless channels, such as frequency hopping and channel blacklisting. However, there are still open issues, such as the appropriate method of managing both mechanisms to avoid similar or adjacent channels from neighboring nodes, which leads to collisions

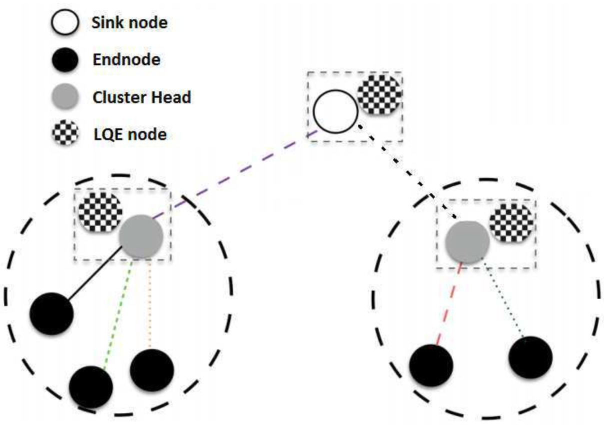

Figure 2: Example of topology used in the thesis, with cluster heads and sink node working as LQEs.

and interference in the spectrum used by the sensors.

1.1 Thesis Statement

The research problem is related to the difficulty of using sensor networks in industry due to the problems faced by the network devices, like interference and multipath fading. The challenge is to achieve a high Packet Delivery Rate (PDR) with a low energy consumption and low delay in such environments.

Not all 16 available channels defined by the IEEE 802.15.4 standard present the same characteristics at the same time, and some of them have at least acceptable quality, and should be used by the sensors, while other low quality channels should be temporarily blocked and used later. On the other hand, with less available channels, the network decreases its diversity and increases the likelihood of collisions and internal interference, especially when the network has many devices with simultaneous transmissions.

Despite the lower diversity and greater likelihood of collisions, the use of channel lists tends to increase the PDR in harsher and dynamic environments. In less troubled environments, usually office and home, with less changes in the network topology, the use of channel lists does not seem to have considerable positive effects on PDR, as is presented throughout this thesis.

Thus, in industrial environments it is necessary to carry out the separation of the channels through different lists according to the profile of each channel at a certain time so that they can be managed in a better way and such problems can be mitigated. The temporary profile of each channel in each link at given times is assessed through Link

Quality Estimator (LQE).

The main idea is to propose a method that manages the lists of channels using an LQE, and that this management considers simultaneous transmissions so that internal interference and collisions can be avoided.

Figure 2 presents the example of an application with tree topology used in the thesis. Besides tree topology, star topology was also evaluated, in which there is a sink node that works as network coordinator and LQE. In tree topology, a network with four cluster heads and 48 end nodes connected to the cluster heads was evaluated. Each cluster head is directly connected to 12 different end nodes. It works as coordinator and router, and forwards the packets from the end nodes to the sink node. In a star topology, 16 end nodes are used to send data packets to the sink node.

A difference between both topologies is that in the star, there is no simultaneous transmission, in spite of the IEEE 802.15.4 standard considering this possibility. In the tree topology, there are two or more devices transmitting in the same timeslot, i.e., at the same time.

In the example of Figure 2, the LQE node is actually a function embedded in the coordinators, but it could also be a separate node close to them, which removes this task from the network coordinators.

In the star topology, each sensor has its timeslot for data transmission, and each transmission is made on a different channel, which facilitates the management of channels, but at the same time decreases the network scalability. Since the sink node receives a single packet for each timeslot, in very large networks the end nodes can take a long time to be allowed to transmit the next data packets. This problem can occur, unless the time schedule allows packets from multiple end nodes to be transmitted in the same timeslot. In this case, it is necessary to develop a mechanism that assigns priorities to each sensor and each package, to guarantee a fair competition for data transmission.

It is a challenging task to configure the channel lists and to schedule the node transmissions so that two or more neighbor links with parallel transmissions do not use the same or adjacent channels at the same time. Although it improves the overall performance, the use of channels lists can decrease network capacity, since the same traffic must be forwarded through a smaller number of channels. However, due to the high variation in the quality of the links, the use of lists ensures that the sensors do not hop to channels with low quality, allowing an increase in the PDR and a reduction in energy consumption due to the reduction in the number of retransmissions.

To create a list, it is necessary to analyze the channels beforehand, and for this purpose, the use of a LQE is required, and the estimation process must be continuous to guarantee a high PDR.

The importance of the correct channel list management has been neglected by most works, which present more studies about scheduling policies, energy efficiency, and multi-hop networks [Queiroz et al. 2017], and perform a blind hopping with no prior assessment of the channel quality. This thesis addresses this issue and proposes a solution that manages the channels in a way that considers the local characteristics of the wireless media, and seeks to assign spectrally distant channels to devices with simultaneous transmissions.

The frequency hopping method is proposed, based on the industrial standard IEEE 802.15.4e, more specifically, the IEEE 802.15.4e TSCH mode, with the assignment of spectrally distant channels for nodes with parallel transmissions through channel offset management. In addition, two proposals are used for channel lists: one that uses two channel lists and another that uses three channel lists. Both solutions showed superior performance when compared to the blind channel hopping approach based on the TSCH protocol, with and without channel lists.

1.2 Objective

The main objective of this thesis is to develop a distributed approach based on the channel hopping method of the TSCH protocol to manage the list of channels to improve the PDR in industrial environments. The objective is to design a collision-free protocol with a local channel listing management through a link quality estimation method to mitigate the drawbacks of interference and multipath fading in IWSN. The first solution uses two channel lists, which is the most common approach in the literature, and the second solution is based on fuzzy logic to define which of the three lists the channels will be inserted into.

The design of such solutions for WSN in industry is demanding. They analyze the channels and list those that are of sufficient quality to be used to guarantee high PDR to the sink node, where they can be processed. None of the neighboring nodes use the same adjacent channels in simultaneous transmissions. The coordinating nodes, which means the sink node and the cluster heads, monitor the channels and define for each local link the most appropriate ones to be used, also considering the problem of simultaneous transmissions when assigning different channel offsets for each subnet.

1.3 Main Contributions

- Experimental studies to evaluate the wireless channel in industrial environments were performed;
- Simulation studies were carried out using a developed channel model that simulates industrial environments, with the objective of evaluating the frequency hopping and channel adaptation methods, identifying the advantages and disadvantages of each one;
- Experimental studies were also performed in a laboratory environment using a single channel, with several channels by means of frequency hopping, and with the use of channel denylists;
- A new link layer protocol is proposed, which uses link quality estimator at the receiving nodes, performs frequency hopping and uses double channel list, which divides the channels according to the quality of each one;
- A similar method of the developed protocol is proposed, however this one uses a triple list of channels by means of a fuzzy logic set, in addition to using link quality estimators and frequency hopping. The triple list allows to better differentiate the quality of each channel.

1.4 Thesis Outline

The remainder of this thesis is organized as follows.

- Chapter 2 surveys the state-of-the-art about the researches of wireless sensor networks focused on industrial environments, including the open research issues, and the channel model developed to simulate the industrial environments;
- Chapter 3 presents an overview of the IEEE 802.15.4e standard, especially the TSCH protocol;

- Chapter 4 presents a proposal of a novel link layer protocol for IWSN based on the TSCH mode called Adaptive-Blacklist TSCH (AB-TSCH) and presents the results of simulated experiments with the developed channel model;
- Chapter 5 presents a proposal of an improvement of the channel denylisting method, using other two channel lists by a fuzzy logic method that helps classify the channels in these lists. The results are also presented and compared with other approaches with and without channel denylists;
- Finally, Chapter 6 presents some research directions for future works.

2 Industrial Wireless Sensor Networks

In this chapter, an overview of the important issues and some research results on IWSN, that were considered in this thesis, are presented. Section 2.1 focus on WSNs for industrial standards. Section 2.2 shows the main aspects of a WSN project and what should be considered when deploying wireless sensors in industry. In Section 2.3, the software and hardware characteristics of a WSN project are presented, and the most used solutions in literature are discussed, with the developed simulation model. The final remarks are presented in Section 2.4.

2.1 Introduction

The WSN nodes are equipped with sensors (or actuators), they have processing capabilities, resource constraints with low processing power and, in some cases, present restrictions regarding power consumption.

In industry, sensors are deployed to monitor critical parameters such as vibration, temperature, pressure and motor efficiency [Delgado Gomes et al. 2013]. The measurements obtained by them are transmitted wirelessly to a sink node, which provides the information for analysis by a monitoring central, or to be used in control systems. In some cases, the acquired data is locally processed (at the end node) before transmission.

A variety of information can be acquired by the WSN for different purposes, which allows taking appropriate decisions. Based on the obtained information, it is possible to fix or replace the equipment before major losses occur.

The use of WSN in industrial systems presents some challenges. Wireless networks use an inherently unreliable communication medium, which can be aggravated due to noise and interference in the spectrum band used for communication. Different types of interference sources for WSN can be found in industrial environments, such as welding equipment, microwave ovens, and other wireless communication devices (e.g. Wi-Fi or Bluetooth networks), as depicted in Figure 3. In addition, in industrial environments there are machinery, and many metal objects and obstructions, which can affect the quality of the wireless channel. Thus, besides noise and interference, the communication channel is affected by heavy multipath propagation effects [Cheffena 2012], which results in a high degree of attenuation in large and small scale [Tanghe et al. 2008].

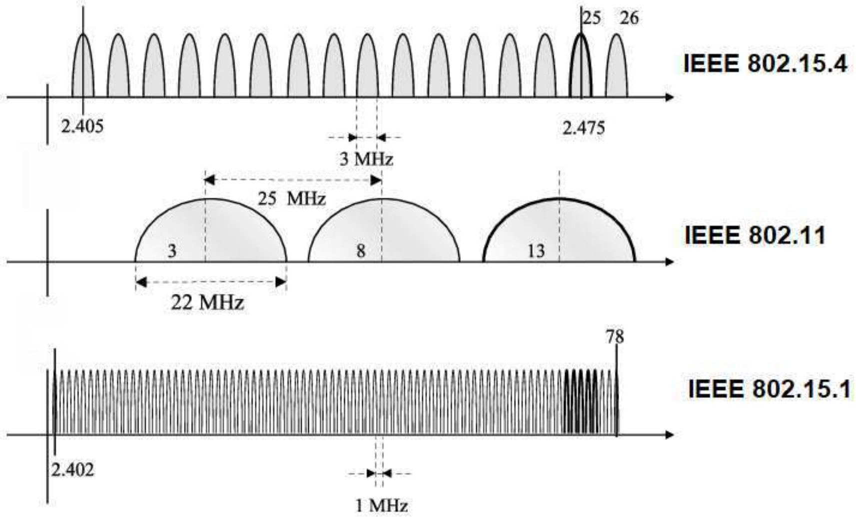

Figure 3: Technologies that can interfere with each other, adapted from [Bertocco, Gamba and Sona 2008].

Compared to other indoor and outdoor environments, the industrial ones are harsher due to the unpredictable variations of temperature, pressure, humidity, and so on. In addition, the wireless channel in many industries is non-stationary in the long term, which can cause abrupt changes in the characteristics of the channel over time [Agrawal et al. 2014].

The lack of reliability makes it difficult to guarantee a certain Quality of Service (QoS). Since the industry intends to reduce the capital and operational expenditure without losing QoS [Kumar S., Ovsthus and Kristensen. 2014], the sensors need to be low cost, resulting in a set of restrictions, such as low rate and low processing capabilities.

Fig. 4 shows a general IWSN configuration, in which the dashed arrows represent the wireless links of the end nodes, and the solid arrows represent the links between the clustering nodes (CH) and sink nodes. The sink nodes may have multiple wireless interfaces, each one using a different channel to allow simultaneous transmissions from the CHs. The end nodes can communicate with a sink node using a single hop or multiple hops through a CH or intermediate routers (nodes illustrated in green). In this scenario, the intermediate routers are nodes that can act both as a sensoring device (end node), and as a router. The Final Processing Node is connected to the sink nodes, and is in charge of collecting and analyzing all the data received from the WSN. Usually the connection between the sink nodes and the Final Processing Node is done using a wired link.

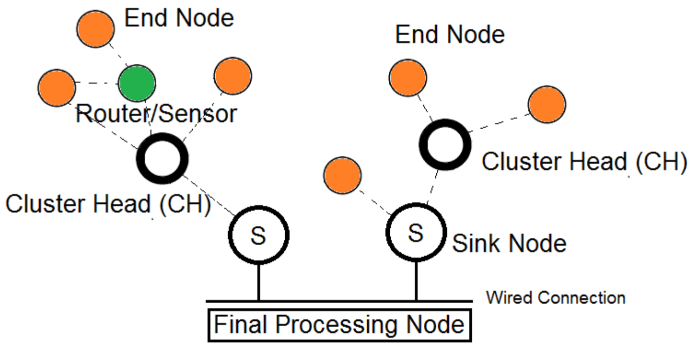

Figure 4: General Wireless Sensor Networks.

The absence of wires makes it practical to perform periodic data collection and event detection, but it is necessary to implement media access protocols that consider QoS, to avoid unacceptable delays when urgent or important messages related to rare or unexpected events are stored in the delivery queue (buffer) and sent through a transmission medium.

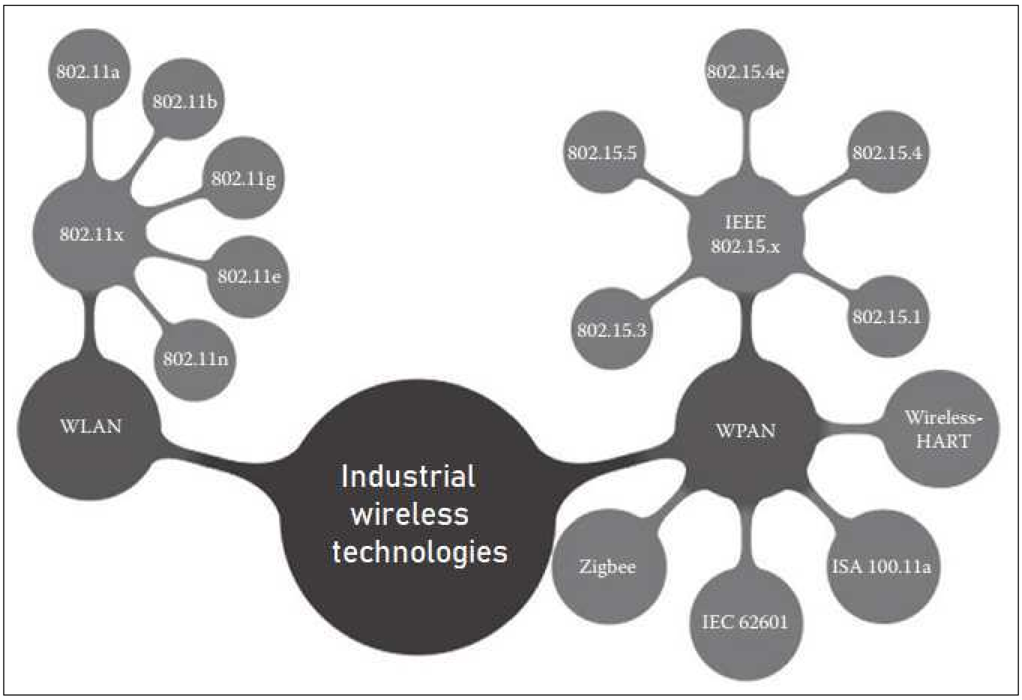

Figure 5: Industrial wireless networking technologies, adapted from [Kazmierkowski 2014].

To deal with the adverse effects inherent to wireless medium in industry, several standards have been developed and are illustrated in Figure 5, in which WirelessHART and ISA100.11a are the most well-known for low capacity devices, with IEEE 802.15.4e as the latest one. ZigBee standard, which is based on IEEE 802.15.4-2003 of the Media Access Control (MAC) layer, was developed for WSN in general purpose, and is still widely used for research in industrial environments, although it has been gradually replaced by more appropriate technologies, such as WirelessHART and IEEE 802.15.4e [Queiroz, Gomes and Benavente-Peces 2017]. Officially unveiled in 2007, WirelessHART remains one of the main standards surveyed for the industry, and due to improvements involving the IEEE 802.15.4 standard, IEEE 802.15.4e has recently caught the attention of researchers.

WirelessHART uses only 15 channels defined in the 2.4 GHz band, since channel 26 is not allowed in some countries [Petersen and Carlsen 2009], and ISA100.11a uses the 16 channels in the 2.4 GHz band (channel 26 is optional).

Zigbee has many disadvantages, and the main drawback is that it does not provide any coexistence mechanism and is based on the MAC protocol of the IEEE 802.15.4 standard, which uses the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) with a single channel. Thus, this technology is not equipped to deal with multipath and interference problems in a satisfactory way, not recommended for use in IWSN [Akerberg, Gidlund and Bjorkman 2011].

In CSMA/CA, sensors transmit only when they detect the idle transmission medium, and remain active for the entire waiting period. Using the Time Division Multiple Access (TDMA) method instead of CSMA/CA has some benefits. In TDMA, collisions in transmissions are avoided, thus reducing the extra energy consumption by the sensors. In addition, this consumption can be optimized, as nodes can turn off the transceiver during the timeslots when they are not transmitting or receiving packets.

WirelessHART and ISA100.11a use the physical layer of IEEE 802.15.4, but define different protocols for the MAC layer based on the TDMA access method, frequency hopping or a combination of both mechanisms. Even defining mechanisms to deal with the lack of reliability, these protocols can still face some problems. For example, when using frequency hopping, nodes often switch to a new channel before each transmission. However, if proper channel list management is not done, network performance can be significantly degraded, as the node is likely to hop to a channel with low quality [Gursu et al. 2016].

The work in [Wagner and Barton 2012] describes a study to verify the performance of ISA100.11a and Zig-Bee Pro radios in the context of aerospace applications. ISA100.11a radios present higher cost if compared to radios compatible with the IEEE 802.15.4 standard, but can perform better in environments with high interference. The experiments

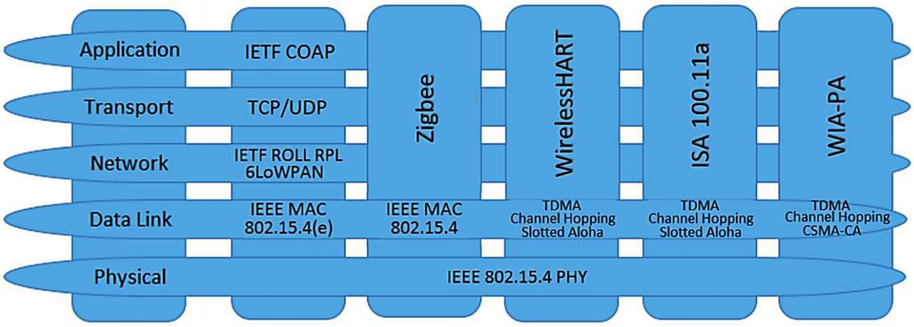

Figure 6: Standardization of wireless technologies with low power consumption and limited resources.

showed that ISA100.11a radios maintained good communication performance even in the presence of interference, while ZigBee radios showed a drop in performance. However, the latency of transmissions in ISA100.11a is much higher.

As briefly mentioned, to overcome the limitations of the IEEE 802.15.4-2003 standard, IEEE created the 802.15 Task Group 4e (TG4e)[1] that redesigns the existing MAC 802.15.4 protocols. The objective was to define a low-power frequency hopping MAC protocol, capable of meeting the emerging needs of industrial applications. This improvement resulted in the document IEEE 802.15.4e MAC, released in 2012. This standard uses many ideas from the WirelessHART and ISA-100.11a standards, including access by shared and dedicated timeslots, and communications through multiple channels. Specifically, IEEE 802.15.4e extends the previous IEEE 802.15.4 standard by introducing five new operation modes. Among them, only TSCH, DSME and LLDN have been explored in the literature, so far [Queiroz, Gomes and Benavente-Peces 2017].

In addition to the standards mentioned, which focus mainly on the MAC layer, other efforts for standardization have been made focusing on other aspects of the network. The Internet Engineering Task Force (IETF) has defined several protocols for integrating smart devices into the Internet, leading to the concept of IoT. Some of the most important are the protocol for low consumption personal IPv6 networks (6LoWPAN)[2], the routing protocol for low power and lossy networks (RPL)[3] and the application protocol restricted (CoAP)[4], which enables applications on smart devices, as illustrated in Figure 6.

The concept of smart devices is closely linked to IoT, and IIoT is the application of these technologies in industrial settings, such as in smart factories. This concept

[1]IEEE TG4e - http://www.ieee802.org/15/pub/TG4e.html - Accessed in 03/05/2020.
[2]IETF 6LoWPAN - https://tools.ietf.org/html/rfc8138 - Accessed in 03/05/2020.
[3]IETF RPL - https://tools.ietf.org/html/rfc6550 - Accessed in 03/05/2020.
[4]IETF CoAP - https://tools.ietf.org/html/rfc7252 - Accessed in 03/05/2020.

expands traditional automation systems and industrial computer systems in a much broader context [Hairong Yan et al. 2014].

As a way of directing this research, the focus of the thesis is related to the development of a MAC layer protocol that can deal more effectively with the inherent problems in wireless sensor networks. To design a WSN with the proposed solution, some aspects must be considered.

2.2 Important Aspects in the WSN Project

The main aspects to be considered when designing a WSN for industrial environments are the use of multiple channels and the use of link quality estimators, which help the sensors to efficiently execute the frequency hopping method.

2.2.1 Multiple Channel Approach

Experiments described in [Amzucu, Li and Fledderus 2014] evaluated the multiple-channel approach, and showed that changing the communication channel can lead up to a 30 dB of difference in the received power inside an office environment. In [Varga et al. 2016] experiments were performed for a short range in an environment without multipath, and with Line-Of-Sight (LOS).

In that experiment, differences up to 10 dB were observed for some channels. In the experiments described in [Watteyne et al. 2010], performed in an office environment, even the adjacent channels were uncorrelated, for distances larger than 6.5 meters. This difference may be larger in channels with high Root-Mean-Square (RMS) delay spread, as can occur in many industrial environments.

Several protocols that implement multiple channels were developed for WSN, however in this topic their classification considers that only the coordinating nodes can have multiple transceivers. Thus, the end nodes are assigned the task of only sensing and transmitting the data, saving energy and ensuring the stability of the network. Multichannel protocols can be divided into static, dynamic and semi-dynamic [Soua and Minet 2015].

2.2.1.1 Static Protocols

In the static protocols, there is no synchronization mechanism or overhead in relation to channel changes. These characteristics make them the simplest to implement, compared to other approaches. Their disadvantage is that they are not able to adapt to variations in the channel characteristics over time, since they are configured only at the beginning of the network.

An example of a static protocol is the Tree-Based Multi-Channel Protocol (TMCP) [Wu et al. 2008]. It divides the network into separate subtrees (or clusters), with the sink node as the root. To construct the subtrees, a width-based search algorithm is applied. Then, orthogonal channels are allocated to each subtree.

The algorithm allows to increase the network flow through simultaneous transmissions in different sub-trees. The disadvantage is that it is not able to eliminate interference between nodes in the same subtree. In addition, dynamic variations in the quality of the channels are not considered, and some assigned channels may have low quality in certain periods, which would prevent the use of these channels at certain times.

2.2.1.2 Dynamic Protocols

In this category, the protocols use several channels simultaneously to reduce collisions by means of scheduling strategies. These strategies prevent the same channel from being used in the same timeslot by nodes with simultaneous transmissions. For this, it is necessary to define synchronization mechanisms to ensure that the transmitter and receiver are on the same channel during transmission. The disadvantage arises when packets are transmitted in broadcast, and when new nodes are added [Incel 2011].

WirelessHART, ISA100.11a and TSCH standards use the dynamic approach. Other example of a dynamic protocol is Y-MAC [Kim, Shin and Cha 2008]. It is a multichannel MAC protocol based on TDMA that avoids assigning redundant channels by not allocating fixed channels to nodes. When bursty traffic occurs, a receiver and potential transmitters hop to one of the other available channels, according to the hopping sequence. If each node has an exclusive time interval for sending in a distance of two hops from the other node, the access to the medium without collision will be guaranteed.

There are also protocols that use an asynchronous approach, with the assignment of a control channel. Its advantage is that there is no need for synchronization between the sensor nodes, however there may be excessive processing in the negotiation of the control channel, as the nodes periodically need to listen to the control channel, waiting for a message indicating the start of a frame for data exchange. In addition, while waiting for control messages, the other channels remain idle. In very dynamic environments, the asynchronous strategy suffers yet another constraint, which is the bottleneck in relation to the control channel, and may present low quality at times.

2.2.1.3 Semi-dynamic protocols

There are several semi-dynamic protocols [Soua and Minet 2015], but the most relevant and studied are MMSN [Zhou et al. 2006] and EM-MAC [Tang et al. 2011]. The Multi-Frequency Media Access Control for Wireless Sensor Networks (MMSN) consists of the assignment of frequencies and aspects of medium access. Different frequencies are assigned or even allocated frequencies available to nodes that have potential conflicts in communications. The protocol allows users to choose one of four frequency assignment strategies available, each addressing different application requirements.

The first strategy of the MMSN is called Exclusive Frequency Assignment, which assigns different frequencies to nodes that are two hops apart if the number of frequencies is equal to or greater than the number of nodes within two hops. The second strategy, called Implicit-consensus, performs the same task but with less overhead in communications, and requiring more available frequencies. The last two strategies, Even Selection and Eavesdropping, work differently and are designed for use when the number of available frequencies is less than the number of nodes within two hops. The difference between them is that Even Selection has a greater focus on conflict reduction, while Eavesdropping is more energy efficient.

The Dynamic Multichannel Energy-Efficient MAC Protocol (EM-MAC) is a multichannel asynchronous MAC protocol initiated by the receiver, which does not use a control channel and does not require the nodes to synchronize their clocks with each other. Instead, each node independently decides its own frequency hopping approach in a pseudo-random fashion, exchanging only a few control information. The protocol EM-MAC is based on contention, hence it is more suitable for networks with reduced data traffic.

In [Algora et al. 2016] the authors made a comparison between EM-MAC and Orchestra [Duquennoy et al. 2015] scheduling algorithm, which is based on TSCH. The results showed that Orchestra presents a superior performance in terms of PDR, while EM-MAC has lower energy consumption.

Orchestra aims to autonomously create and maintain the transmission time schedule of the nodes using information from the RPL routing tree. The schedule consists of several slotframes (repetitive set of timeslots), each one dedicated to a type of traffic. The types of traffic are application, RPL protocol signaling, and Enhanced Beacons (EB)[5]. Despite the advantages of Orchestra, it does not use a channel denylist nor does it provide for the use of link quality estimators.

In this category of protocols, channel switching and simultaneous transmission on different channels are allowed, but switching occurs less frequently. In some approaches, the receiver node is associated with fixed channels, and the transmitter switches to the channel of its receiver when there are packets to transmit. In that case, a synchronization mechanism may also be necessary if the receiver changes the channel over time.

In [Saifullah et al. 2014] the authors propose three approaches to channel allocation, to minimize conflicts between simultaneous transmissions. The first is based on the receiver, in which the channels are allocated to the receivers on the network. The second is based on the connections between the nodes, in which the channels are allocated for each link between two nodes, so that the same node can receive packets through different channels, from different transmitters. Both aim to make the network free from channel conflicts, without interference due to simultaneous transmissions on the same channel.

In both approaches, the network is modeled as a graph, and an interference graph is presented, in which the edges represent nodes that interfere with each other. After modeling the interference graph, a graph coloring algorithm was used to minimize the number of channels to be used, in which the channels are modeled as the colors in the algorithm.

The third approach aims to minimize the maximum level of interference at the nodes, and also uses an interference graph. The strategy performs static and distributed channel allocation to reduce collisions. After that, the protocol can work dynamically, when a connection-based approach is used. However, this technique does not have mechanisms to adapt to the dynamic variations of the network.

To maintain a certain level of QoS for long periods, it may be necessary to employ dynamic channel allocation strategies. In addition to the ability to choose the best channels, the mechanism must allow a quick synchronization of the network in case of channel change. The allocation must consider the network topology to prevent neighboring subnets from using the same channels or adjacent channels, and some mechanism to estimate the quality of the channels, as there may be spatial variations in quality.

[5]EB is a TSCH frame that may contain information on synchronization, channel hopping and timeslot used in the advertised network.

2.2.2 Link Quality Estimation

Channel quality estimators are the basis for various routing protocols, topology control and dynamic channel allocation mechanisms [Baccour et al. 2012]. An example of a routing protocol that uses a quality estimator called CTP, which considers the asymmetry of the channels, is presented in [Gungor and Korkmaz 2012].

Estimators must have good reactivity to changes and maintain good stability. Some of them are based on hardware using the Link Quality Indicator (LQI) and Received Signal Strength Indication (RSSI), others on software (PDR, number of retransmissions, and so on). All packets received by IEEE 802.15.4 radios have an associated RSSI and LQI value, however, RSSI can also be obtained regardless of packet reception. In this case, the transceiver performs a power scan of the channel, regardless of the source. Any device that generates noise and interference influences its value. The LQI, on the other hand, can only be measured during the reception of packets, since its metric is based on the analysis of the first symbols of packets received. RSSI is more stable than LQI, except in very reflective environments, which show larger attenuation on a small scale due to fading over multiple paths [Baccour et al. 2012].

Estimators based on software use information obtained from the upper layers, such as PDR, and the Required Number of Packet Transmissions (RNP). The use of metrics based on PDR allows a good estimate of links with very high or very low quality, but presents some problems in links with intermediate quality. When retransmission is used, PDR-based metrics can overestimate the quality of the link, as it does not consider the number of transmission attempts before a successful reception. RNP-based metrics estimate the required number of packet transmissions until successful reception. ETX and Four-Bit (FB) are examples of estimators based on RNP [Gomes et al. 2017].

The ETX [De Couto et al. 2003] estimator is initiated by the receiver, and when estimating the PDR in both directions, it considers the link asymmetry. To calculate the PDR of the links on the two routes, broadcast probe packets (probe broadcast) are used, causing overload on all nodes in the network, in addition to extra traffic.

The FB [Fonseca et al. 2007] estimator is initiated by the sender and uses four bits of information. The first bit is obtained from the physical layer to identify the quality of the channel in a received packet. The second bit (ACK) is obtained from the link layer and considers the links in both ways. The last two bits are obtained from the network layer and are useful for route decisions [Baccour et al. 2012]. The bit ACK is determined using data packets and broadcast probe packets that are combined to calculate an ETX estimate.

The F-LQE [Baccour et al. 2015] estimator is initiated by the receiver and is based on the fuzzy logic. It uses four metrics to characterize the link: packet delivery (SPRR), stability (SF), asymmetry (ASL) and quality of the channel, which can be the signal-to-noise ratio (SNR) or the LQI. The first metric is the PDR filtered with a filter called Exponentially Weighted Moving Average (EWMA). The stability metric is the coefficient of variation of the PDR. The third metric uses the PDR calculated on neighboring nodes, and these values on each node are acquired using broadcast probe packets and sent along with data packets. The SNR calculation is performed using two RSSI values, one sampled from a received packet, and the other sampled after receiving the packet to obtain the noise level. After that, both values are subtracted to obtain the SNR.

The Opt-FLQE [Rekik et al. 2015] estimator is a modification of F-LQE to improve reactivity and reduce computational complexity. It uses a metric obtained from the sender, the Smoothed RNP (SRNP), instead of SF. SRNP values are sent along with data packets to allow the receiver to calculate the Opt-FLQE metric.

Another solution for LQE, called Lightweight Packet Error Discriminator (LPED) [Barac et al. 2014], uses Forward Error Correction (FEC) to perform channel diagnostics and information obtained from received data packets. The discriminator LPED is able to identify fading or interference problems on the wireless channel, but does not provide metrics to assess link quality in terms of a network performance indicator, such as PDR or RNP [Eskola and Heikkilä 2016]. This can lead the network to make changes to routes or channels due to rapid variations in the quality of channels, which causes protocol instability. The use of FEC can also be useful to recover some corrupted packets, but when good channels or routers are chosen, the SNR is increased, so the use of FEC may not be advantageous.

In [Gungor and Korkmaz 2012] a simulation study was carried out to compare PDR, RNP, WMEWMA, ETX and FB in the CTP protocol for smart grid environments. The results showed that ETX and FB performed better in harsh environments such as the industrial one, since only the two estimators consider the link asymmetry between the link quality estimators evaluated. In the experiments described in [Baccour et al. 2015], F-LQE outperformed ETX and FB, and in [Rekik et al. 2015], a performance analysis showed that OptFLQE is more reactive than F-LQE, being even more reliable for smart grid environments. In [Rekik et al. 2016], Opt-FLQE was compared to ETX and FB in the RPL routing protocol for smart grid environments, and the simulations showed that Opt-FLQE outperformed ETX and FB for all metrics evaluated.

In [Gomes et al. 2017], the authors proposed an LQE for IWSN, and a new type of node, the LQE node, which estimates the quality of links in real time using RSSI and information obtained from received data packets. The proposed LQE is capable of capturing the effects of multipath, interference and asymmetry of the links.

The experiments were carried out in a real industrial environment using IEEE 802.15.4 radios, and models were developed to allow the use of RSSI samples to estimate the quality of the links. A comparison was made with the Opt-FLQE, and the results showed that the estimator described is more accurate and reactive for the type of environment under study. Unlike other LQEs in the literature, in the proposed LQE, sensor nodes do not need to send test packets in diffusion. In addition, using the LQE node, the other network nodes do not need to interrupt their operation to monitor the quality of the link. In this solution, no overhead is imposed on the final nodes and there is no dependence on the size of the packets, as in LPED, for example.

This work aims to use the idea of LQE in the controlling devices, such as the sink node and the CHs, to estimate the quality of the channels and insert them into the appropriate channel list. The LQE considers the asymmetry of the links, uses RSSI, the number of retransmissions and the PDR to estimate the quality of the channels.

2.3 Software and Hardware for WSN

Sensor nodes are usually composed of one or more sensing units, a power supply operated with battery, or connected to the power distribution company, a processing unit used to process, store, encrypt and digitally modulate, a transceiver and an antenna, as illustrated in Figure 7 (a). In the case of devices such as the LQE node, it may have several antennas to capture information from several channels at the same time.

Connected to the hardware of the sensor nodes, there are the following software subsystems, as illustrated in Figure 7 (b):

- Operating System (OS): designed to run on devices limited in memory, power, bandwidth, and processing capacity. Contiki, OpenWSN and TinyOS are three examples of this type of operating system. They are open source, and connect small, low-cost, low-power microcontrollers (MCU) to the Internet. Table 1 describes the

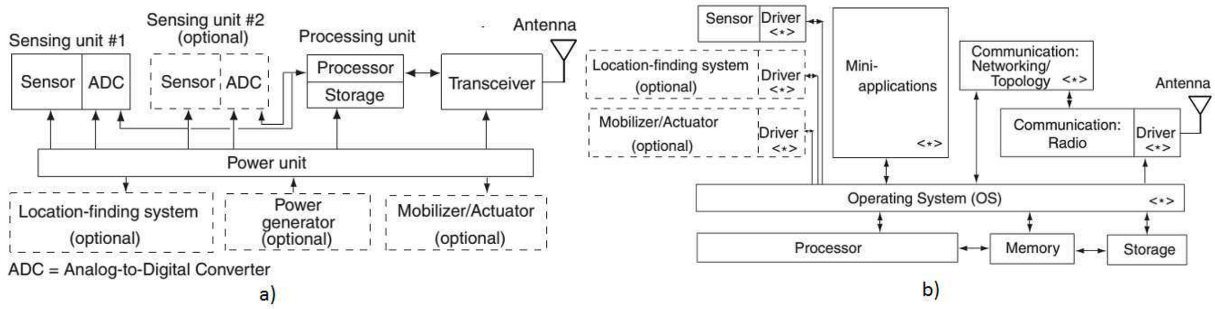

Figure 7: Hardware (a) and Software Components (b), adapted from [Sohraby, Minoli and Znati 2007].

Name	Architecture	Scheduler	Model	Supported MCU	Language	License
TinyOS	Monolithic	Cooperative	Event-driven	AVR, MSP430	nesC	BSD
Contiki	Monolithic	Cooperative/ Preemptive	Event-driven, Protothreads	AVR, MSP430, ARM	C	BSD
OpenWSN	Monolithic	Cooperative	Event-driven	MSP430, ARM Cortex-M	C	BSD
RIOT	Monolithic/ Layered/ Microkernel	Preemptive, tickless	Multi-threading	AVR, MSP430, ARM7, ARM Cortex-M, x86	C, C++	LGPLv2
FreeRTOS	Microkernel	Cooperative/ Preemptive	Multi-threading	AVR, MSP430, ARM, x86	C	Modified GPL

Table 1: Operating Systems.

operating systems for devices with limited resources;

- Drivers: the sensor and transmission drivers are the software modules that manage the basic sensor functions and the details of the radio channel transmission link, including the clock and synchronization methods, signal encoding, bit recovery, bits counting, signal levels and modulation transceivers;

- Mini-applications: data processing, signal value storage and applications;

- Network processors: manage transmission functions, including routing, buffering and packet forwarding, topology maintenance, media access control and encryption.

Wireless sensor processors are commonly referred to as MCUs, and some of the features by which these MCUs are especially suitable for embedded systems are their flexibility in connecting to other devices (such as sensors), their set of instructions enabled to capture and process signals with critical time, and its typically low energy consumption. In addition to these features, the MCU has the ability to reduce energy consumption by entering a sleep state in which only parts of the controller are active.

The most used MCUs in wireless sensor prototypes include the processor from Atmel and Texas Instruments. Table 2 shows some examples and characteristics of the MCU from both manufacturers.

Platform	mica2	openmote-b	micaz	tmote sky	telosB
MCU	Atmel Atmega 128L	TI ARM Cortex-M3	Atmel Atmega 128L	TI MSP430	TI MSP430
Frequency	8 MHz	32 MHz	7.37 MHz	8 MHz	8 MHz
RAM (Kbytes)	4	32	4	10	10
Memória Flash	128	512	128	48	48
Radio	CC1000 315/433/ 868/916 MHz 38.4 Kbauds	CC2538 2.4 GHz 868/915 MHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4
Operating Systems	TinyOS	OpenWSN Contiki	TinyOS	TinyOS Contiki	OpenWSN TinyOS Contiki

Table 2: Platforms and features for WSN.

It is possible to see that some platforms present dual band operation (2.4 GHz + 868/915 MHz), and Table 3 shows the information about the physical layer of IEEE 802.15.4 standard and the frequency spectrum. The 868/915 MHz ISM bands could be used as an alternative to mitigate the problems with interference, since they are known to be relatively interference-free, but several radio technologies have proven to cause significant problems to deployed WSN in 868 MHz band [Barrenetxea et al. 2008] and 900 MHz frequencies [Kusy et al. 2011].

Although the chosen physical layer depends on local regulations and user preference, for the purposes of this thesis only the higher data-rate, worldwide, unlicensed 2.4-GHz band was considered.

2.3.1 Operating Systems

This section describes the most well-known and used Operating Systems (OS) in WSN, with emphasis on Contiki and OpenWSN, which support the device defined for the experiments in this project.

Frequency spectrum	Modulation	Bit Rate	Symbol Rate	Number of channels
868 MHz	BPSK	20 kbit/s	20 kbaud	1
915 MHz	BPSK	40 kbit/s	40 kbaud	10
2.4 GHz	O-QPSK	250 kbit/s	62.5 kbaud	16

Table 3: Physical layer of IEEE 802.15.4

TinyOS is a Berkeley Software Distribution (BSD) licensed OS designed for low power wireless devices, and is one of the first OS designed specifically for WSN. Its architecture consists of a scheduler and a set of connected components [Dargie and Poellabauer 2010].

In TinyOS, scheduled tasks are based on the FIFO principle, and the TinyOS architecture is effective for short tasks. Resource allocation in TinyOS is optimized by adopting a static memory allocation, which avoids the extra overhead associated with dynamic allocation.

TinyOS has two standard implementations for communications between sensors: the first is the Active Message (AM) protocol and the second is the TKN15.4 protocol, an IEEE 802.15.4-2006 MAC platform-independent implementation, developed by the Telecommunication Networks (TKN) Group of the Technical University of Berlin in 2009. A performance analysis of both protocols with sensors based on the TelosB platform was done in [Queiroz, Gomes and Benavente-Peces 2017], and the results showed that TKN15.4 surpasses AM in terms of energy efficiency and PDR. The advantage of AM is that it allows to run several services using the same radio, but at the cost of excessive energy consumption.

Based on an analytical study conducted in [Basmer, Schomann and Peter 2011], it has been shown that TKN15.4 is the most efficient implementation at runtime, compared to other implementations, such as open-ZB, TIMAC, IHP MAC SDL protocols, and IHP MAC cb. The analyzes measured the execution time of a set of MAC operations in which the TKN15.4 presented the shortest execution time. TKN15.4 brings an elegant solution to mitigate time-critical operations in the version with the transmission mode of beacons enabled [Basmer, Schomann and Peter 2011].

According to [Amjad et al. 2016], TinyOS is the most suitable OS to operate on a network with few resources, such as WSN. A survey, that focused on industrial environments, identified that TinyOS is, by far, the most popular OS in IWSN because of its support for various platforms [Queiroz, Gomes and Benavente-Peces 2017].

The operational system RIOT, designed for real-time operations, is based on a micro-kernel with support for multi-threading [Will, Schleiser and Schiller 2009]. It has several network stacks, including its own implementation of the 6LoWPAN stack (GNRC stack), also the OpenWSN 6TiSCH[6], and the network stack called Lightweight Content Centric Networking (CCN-lite) [Hahm et al. 2015].

[6] IPv6 over the TSCH mode.

Contiki is an open source OS designed specifically for WSN. Its kernel is event-driven like TinyOS, and the system includes protothreads which provide a programming abstraction similar to a thread, but with little memory overhead. Protothreads in Contiki are a combination of some event resources and threads.

Contiki has implementation of the Internet Protocol (IP) protocol, with both IPv4 and IPv6 addressing. It uses μIP (*micro IP*), which is an open source implementation of the TCP/IP network protocol stack intended for use in MCU of 8 and 16 bits. μIP is used by hundreds of companies in systems such as cargo ships, satellites and oil drilling equipment[7].

Contiki includes several extensions (add-ons) and libraries that provide functionality for communications [Watteyne et al. 2012]. The most relevant is ContikiMAC [Dunkels 2011], a MAC protocol based on CSMA/CA that activates devices periodically to listen to packet transmissions from neighbors. The μIPv6 library provides routing functionality with RPL and 6LoWPAN protocol. The transport layer implements UDP and a lightweight version of TCP. Contiki also implements CoAP, similar to OpenWSN.

In addition to Contiki, there is an updated branch called Contiki-NG (Next-Generation), which is specifically geared to the context of IoT, and provides support for the ARM Cortex-M3 and M4 microcontrollers used in this project, and the MSP430 from Texas Instruments.

Another example of OS is Berkeley OpenWSN [Watteyne et al. 2012]. It is an open source stack that aims to implement low-power wireless standards such as IEEE 802.15.4e TSCH, 6LoWPAN, RPL and CoAP. Among the platforms that can run OpenWSN are GINA (AT86RF231), TelosB (CC2420), LPC (AT86RF231), K20 (AT86RF231) and OpenMote (CC2538).

In [Boccadoro et al. 2018] the authors carried out a comparative study between Contiki and OpenWSN using the TSCH protocol with TelosB, Zolertia Z1 and OpenMote platforms. The experiments measured memory management, energy consumption, synchronization time, duty cycle, end-to-end packet delay and packet loss rate. The results showed that OpenMote offers the best computational and memory resources, and presents lower values for almost all the parameters mentioned above. However, in terms of energy efficiency, TelosB and Zolertia Z1 presented lower energy consumption.

[7]The Contiki Operating System - http://contiki.sourceforge.net/docs/2.6/ - Accessed in 07/05/2020.

2.3.2 Network Simulators

There is a large number of network simulators that have been developed for various types of environments [Wehrle, Günes and Gross 2010]. The top five simulators that support low power device simulation, also called Low-rate, Wireless Personal Area Network (LR-WPAN), are described in this section. They are widely used in the literature because of the resources provided.

The main simulators are TOSSIM, COOJA, OpenSim, NS2 and NS3, and OMNeT++. Regarding OMNeT++, this work pays special attention to one of its frameworks, Castalia[8], due to its realistic model based on CC2420 and CC1000 radio transceivers, and the wireless channel model we developed to simulate IWSN.

TOSSIM is a simulator with the specific objective of running TinyOS applications transparently [Wehrle, Günes and Gross 2010]. The TinyOS component-oriented software architecture supports integration into a simulator, in which a hardware abstraction is provided by software components with specific interfaces, which are applied at run time compilation. The simulation is event-driven and radio communications are modeled with error rates of bit for each unidirectional link. The radio model is quite simplistic, not accurately describing the interference caused by simultaneous transmissions, and it also does not have mobility models.

COOJA is an WSN simulator derived from the Contiki OS, and is based on Java, which executes code by calling methods in native language with the Java Native Interface (JNI) of the Java environment for a compiled Contiki system. COOJA provides OS emulation and instruction set emulation in a single structure. MSPSim [Eriksson et al. 2009] is used as an instruction set simulator and is also written in Java.

Contiki simulator supports the MSP430 microcontroller and includes some hardware peripherals, such as sensors, communication ports, LEDs and sound devices.

OpenSim is a simulator and emulator for WSN that allows to build the OpenWSN protocol stack and applications, and emulate a complete network on Windows or Linux. Each emulated sensor (compiled C code) runs as a process on the computer and communicates with the simulation core in Python language.

The simulation core is a discrete event simulator that contains a timeline, consisting of several events and their codes to be executed later.

[8]https://github.com/boulis/Castalia. Accessed: 14/05/2020.

NS2 is a network simulator oriented to objects and events whose simulations are performed in C++ language and oriented to Tcl objects (OTcl). NS3 simulator is an improvement over NS2 and is developed in C++ like its predecessor, but OTcl was withdrawn in favor of C++ (network models) and the Python language (optional).

NS3 is a robust simulator developed for Internet systems, but has a lesser focus on the lower layers of the protocol stack, compared to Castalia, derived from the OMNeT++ simulator.

The Objective Modular Network Testbed (OMNeT++) simulation environment is an open source discrete event tool based on C++ and used for simulations of communications networks, multiprocessors and various distributed systems. It consists of modules that communicate with each other through messages. Simple modules can be grouped together to form more complex composite modules. A user defines the structure of a module using the OMNeT++ topology description language called NED.

OMNeT ++ is often referred to as a discrete event simulator, but in fact, it is a software that supports various models and simulation structures (frameworks), such as the INET Framework, MiXiM and Castalia. The frameworks are developed independently of the simulation framework and follow their own release cycles.

- INET Framework is an open source network simulation package that contains models for various Internet protocols: UDP, TCP, SCTP, IP, IPv6, Ethernet, PPP, IEEE 802.11, MPLS, OSPF and others. The INET Framework also contains emulation resources;

- MiXiM supports wireless and mobile simulations. The simulator provides detailed models of the wireless channel (fading, etc.), wireless connectivity, mobility models, models for obstacles and many communications protocols, especially at MAC layer. In addition, it provides a friendly graphical representation of wireless and mobile networks and supports debugging;

- Castalia is a simulator for WSN, for medical sensing (Body Area Networks) and, generally, for low power embedded device networks. Researchers and developers can use this simulator to test their algorithms and protocols distributed over a realistic wireless channel with the CC1000 and CC2420 radio models. Castalia uses the log-normal shadowing model as one of the ways to simulate the average path loss. It also models the temporal variation of the path loss as an effort to capture fading phenomena in very unstable environments.

Considering simulators focused on industrial environments, some studies described simulation studies of IWSN [Du, Navarro and Mieyeville 2015, Alderisi et al. 2015], but none of them considered the non-stationary characteristics of the wireless channel for long periods, nor did they consider the correlation in different channels.

A description of a simple and reliable model to simulate multichannel protocols for IWSN, that captures the effects of fading, shadowing and the non-stationary characteristics of the channel, is presented in [Gomes et al. 2017]. It was integrated into the Castalia simulator by the author of this work, and also considers the differences in the characteristics of the different channels and the asymmetry of the links.

2.3.3 Simulation Model for IWSN

In industry, changes in the topology of the environment, such as the movement of a large metallic structure, can cause changes in the characteristics of the channels over time, leading to a difference in the average value of the received power, even with static nodes.

The wireless channel in industrial environment can remain for several hours with the same characteristics, and after that period a sudden change in the channel may occur [Agrawal et al. 2014]. In this section, the channel model developed for simulating protocols with multiple channels for IWSN is briefly presented. This model was integrated into the Castalia simulator, which by default uses a single channel and does not consider the non-stationary characteristics of the channel over long periods.

Castalia is a discrete-event simulator developed in C++, based on the open-source simulation framework OMNeT++. Its radio module is based on real radios for embedded low-power devices, including support to the transceivers CC2420 and CC1000, and log-normal shadowing model, which provides precise estimates for the average path loss. Castalia is also more specific for low-rate networks than other simulators, such as Network Simulator 3 (NS3) and OPNET, which focus on general networks.

At the physical layer, Castalia uses real parameters of IEEE 802.15.4 radios, and a model based on log-normal shadowing. The simulator also allows temporal variation in the received power using predefined samples that simulate the behavior of a channel subject to multipath fading. However, by default the model implemented by the simulator considers that the channel is stationary, i.e., its characteristics remain unchanged during all the time.

To perform the integration with Castalia, modifications were made in the *Wireless-Channel* class, to capture the variations in the channel characteristics over the long run. The average duration for changing the channel parameters to simulate variations in channel quality seemed to be consistent with the experimental results of [Agrawal et al. 2014] and [Wang and Yao 2013], in which the channel remains stationary for some hours. The parameter values for using the log-normal shadowing and Rice models were obtained through experiments in industry described in [Tanghe et al. 2008], considering a scenario without LOS between transmitter and receiver.

To model the effect of fading over a long period, an approach based on the two-state Markov chain was used. This chain consists of a discrete stochastic process based on integer values. In general, the random variables that define a stochastic process are not independent, and may have complex dependency relationships. Markov's processes, on the other hand, present a simple form of dependency. A stochastic process is called a Markov process if it satisfies a property called a Markov property. It shows that the probability of a given event occurring in $k+1$ time depends only on the value of the process in time k. Markov chains can be represented graphically, as a non-deterministic state machine, and is illustrated in Figure 8.

An average time of change (Tc) is defined for the model, in minutes, which is used to define the change probability p. In the implementation of the model, the interval of Markov chain's state change was set to be equal to one minute, which results in $p = 1/T_c$. With this parameter, it is possible to simulate environments that remain unchanged for a long term, as well as environments in which more frequent changes in topology occur. The Rice distribution was used to model the multipath attenuation, but other distributions could be used, such as the Nakagami-m distribution. Different models for attenuation can be easily integrated into the simulation model.

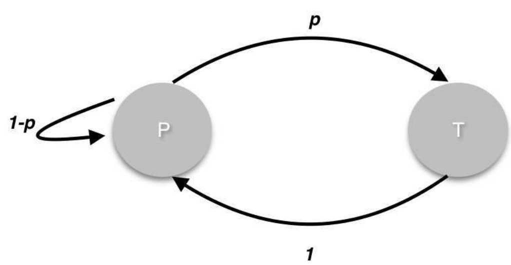

Figure 8: Two-state Markov chain.

Figure 8 shows the two-state Markov chain that remains in the P state as long as the channel characteristics are unchanged. The transition to the state T occurs with probability p. When this occurs, the parameters of the channel models are modified, so that there is a sudden change in their characteristics. When changing the channel parameters, the chain returns to state P with probability one. To define the frequency with which changes in channel characteristics occur, the value of p is established, making it possible to simulate environments that remain unchanged for a long time and environments that present changes in topology more frequently.

As the channels in industrial environments are uncorrelated in frequency, to simulate protocols that use multiple channels, one instance of the model can be used for each channel available for communication. In the studied scenario, changing channels can improve the quality of service of the network, since the network is able to adapt to variations in channel quality. To allow the evaluation of multichannel protocols with Castalia, it is necessary to make modifications to the interface between the module that implements the wireless channel model and the modules that implement the upper layer protocols.

This model is described in detail in [Gomes et al. 2017], and to test it initially, random values were generated using a C++ library called IT++[9] that performs the simulation of received power values at a receiver, with distance $d = 20$ m considering the values of $n = 1.69$, $d0 = 15$ m, $L(d0) = 80.48$ dB and $\sigma = 8.13$ dB, which are applied in the calculation of large fading scale and shadowing, using the log-normal shadowing model. These values were obtained from experiments in an industrial environment described in [Tanghe et al. 2008].

The first step in redefining channel characteristics, when the Markov chain reaches state T, is to define the average reception power ($P_R(d,t)_{\mathrm{dBm}}$) for a distance d and in time t, according to

$$P_R(d,t)_{\mathrm{dBm}} = P_T - L(d). \tag{2.1}$$

In the equation, P_T is the transmission power. $L(d)$ has a deterministic part, related only to the distance between transmitter and receiver, and a random part, referring to log-normal shadowing in Equation 2.2.

$$L(d) = L(d_0) + 10n\log\left(\frac{d}{d_0}\right) + X_\sigma. \tag{2.2}$$

The Gaussian random variable X_σ has a mean of 0 dB and standard deviation σ (in dB). Thus, considering the same distance between transmitter and receiver, the value of

[9] IT++ Documentation - http://itpp.sourceforge.net - Accessed in 08/05/2020.

$P_R(d,t)_{\text{dBm}}$ presents a random variation referring to X_σ, which can increase or decrease the average value of received power.

After defining the average power value in dBm, the corresponding power value in mW is used as a parameter for the Rice distribution, which is sampled to define the power value received at each instant.

Besides large-scale path loss and shadowing, the small-scale attenuation also needs to be considered, due to rapid changes in the multipath profile of the environment caused by the movement of objects around the receiver and transmitter.

Experiments have shown that in industrial environments the temporal attenuation follows the Rice distribution [Tanghe et al. 2008]. This distribution models the small-scale fading when there is a dominant stationary signal and random components overlapping with the main component. In industrial environments, there are usually several invariant rays, and just a small portion of multipath profile is affected by moving objects [Tanghe et al. 2008]. The Castalia simulator does not implement small-scale fading with the Rice distribution, or any other probability distributions.

The probability density function that describes the envelope of a signal under Ricean fading is

$$p_R(x) = \frac{x}{b^2} e^{-\frac{(x^2+A^2)}{2b^2}} \mathrm{I}_0\left(\frac{Ax}{b^2}\right), x \geq 0, \qquad (2.3)$$

in which $\mathrm{I}_0(\bullet)$ is the modified Bessel function of the first kind with order zero, A^2 is the power of the dominant stationary signal, and $2b^2$ is the average power of the random multipath components.

Equation 2.3 shows the probability distribution that models the value of power received at each instant, considering the value of the average power P_R and the variance $\sigma(t)$ in time t. The variable $P_R = 10^{\frac{P_R(d,t)_{\text{dBm}}}{10}}$ is the average power in mW for a distance d and in time t.

For this test, the standard deviation value of the Rice distribution was sampled from a uniform distribution between 0 and 1.5. Thus, small-scale attenuation also behaves differently with each change of state. This type of behavior was observed both in experiments carried out by the authors and in other works [Agrawal et al. 2014].

Table 4 shows the parameters used for the simulation, that uses lognormal shadowing and Rice distribution. The parameters n, d_0, $L(d_0)$, and X_σ are used to calculate the path loss and shadowing, and the parameter K is used to calculate the small-scale fading with Rice distribution. To model the variations in the level of small-scale fading, it is also

Transmission rate	1 packet/s
Transmission power	0 dBm
Average change time	30, 80 and 100 minutes
Path Loss Exponent (n)	1.69
Reference distance (d_0)	15 meters
Path loss at reference distance ($L(d_0)$)	80.48 dB
Standard deviation of shadowing (X_σ)	8.13 dB
Rice Factor (K)	12.3 dB
Standard deviation of the Rice factor (K_σ)	5.4 dB

Table 4: Parameters used in the simulations.

possible to configure a value for the standard deviation of K factor (K_σ). Thus, every time a change in the channel characteristics occurs, a new value of K is defined.

The parameter average change time shows the mean time between two modifications in the channel characteristics in minutes. In [Agrawal et al. 2014], the channel remained stationary for a long term (up to some hours) before the occurrence of an abrupt change in the channel characteristics. For the simulations, three different change times were set to compare the performance of the proposal in very dynamic environments.

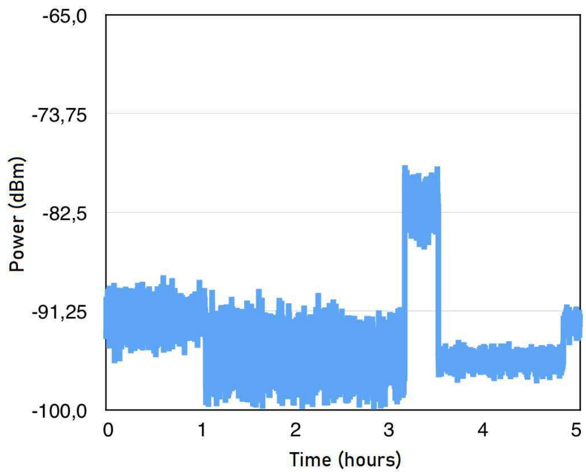

Figure 9: Power received during the simulation.

Figure 9 shows the power received at the node during five hours of simulation. It is possible to observe the moments when variations in the characteristics of the channels occur, both in relation to the average power value, and in relation to the severity of the attenuation by multipath.

Figure 10 shows the PDR during five hours. It is possible to observe a correlation between the PDR and the characteristics of the channel at each moment. This type of behavior is realistic in dynamic environments, which cause variations in the quality of the channels over time, as is the case in industrial environments.

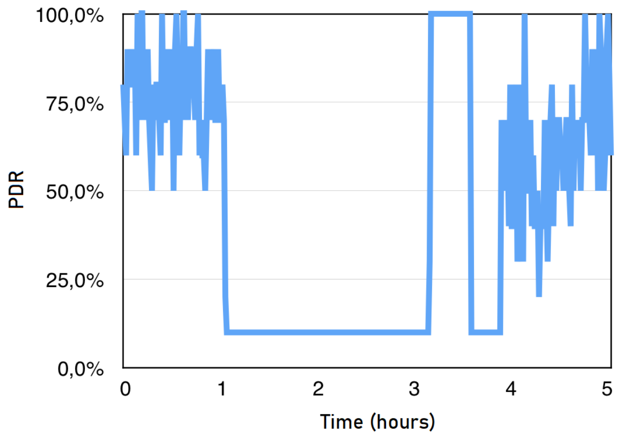

Figure 10: PDR during the simulation.

It is possible to observe that the model is capable of capturing the behavior of the channel over long periods, considering the non-stationary profile of the wireless channel in industrial environments. The results obtained by the model are compatible with the results of experiments carried out in an industrial environment and described in articles that analyzed for long periods [Agrawal et al. 2014, Wang and Yao 2013].

2.4 Final Considerations

This chapter discussed the most widely used IWSN technologies, some of which are not recommended for this type of environment. ZigBee is the best known of the technologies and used in the literature in different environments, but due to the fact that it communicates through only one channel with the CSMA/CA contention access method, this technology does not meet the demanding requirements of industrial environments.

WirelessHART and ISA100.11a are among the most used technologies for WSN, however the former performs denylist management manually, and the latter postpones packet transmission when it finds a denylisted channel, leading to packet delay.

Important aspects to be considered in the design of an IWSN were also discussed, which are the use of multiple channels, guaranteeing diversity, and the use of quality estimators, which guide the use of good quality channels.

In addition to the technologies, the most used software and hardware solutions in IWSN research were studied, with special attention to the Contiki and OpenWSN, in addition to the OpenMote-B platform, supported by the mentioned operating systems, and will be used in future works. Finally, the simulator used in the project was discussed with the developed channel model.

The next chapter presents the IEEE 802.15.4e standard, with special focus on the TSCH behavior mode.

3 IEEE 802.15.4e standard

3.1 Introduction

Several studies have investigated the performance of IEEE 802.15.4, and they highlighted the limitations of this standard, which is not suitable for applications that have strict latency and reliability in hostile environments such as the industrial one [Khanafer, Guennoun and Mouftah 2014], [Baronti et al. 2007], [Tennina et al. 2013], [Zhan, Xia and Anwar 2016], [Du, Navarro and Mieyeville 2015], and [Huang, Pang and Hung 2009].

To overcome the limitations of the IEEE 802.15.4 standard, IEEE 802.15.4e five modes of operation were developed and defined, also called protocols, namely: TSCH, DSME, LLDN, AMCA and BLINK. For the latter two, the standard provides only a brief description, so it is not discussed in this chapter.

In general, improvements to the IEEE 802.15.4e standard relate to support for multi-channel communications, more flexible superframe[1] (DSME) and the use of a TDMA-based contention-free access mechanism, which reduces collision and minimizes energy consumption. In the following section, details of the main modes of operation are described, including the channels denylist management methods, which have been proposed in the standard, but are an open question, since there is no definition of how the management should be performed.

The concepts of channel blacklisting and whitelisting are very common and used by several works [Peng Du and Roussos 2013, Tavakoli et al. 2018, Gomes, Watteyne and Krishnamachari 2017, Kotsiou et al. 2017, Zorbas et al. 2018], however, in this thesis the terms denylisting and allowlisting are used as an effort to adopt inclusive language.

[1]Period between sending the first beacon, contention access period (CAP), contention-free access period (CFP), and the sending of the second beacon.

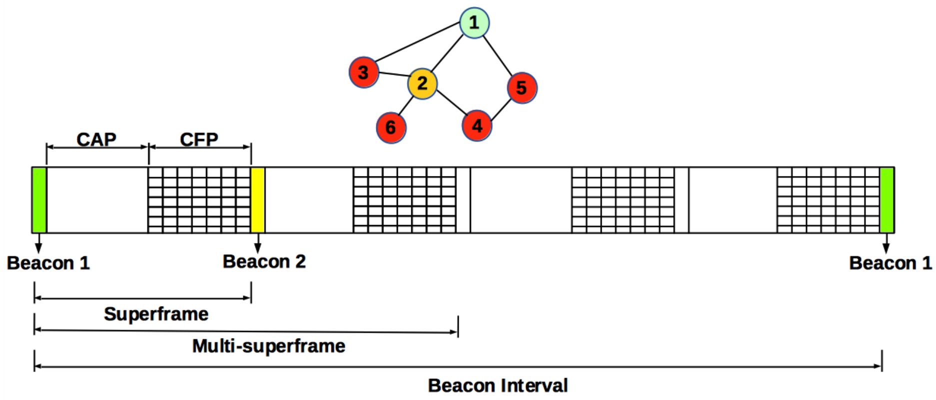

Figure 11: Example of a superframe structure for a DSME network.

3.2 Deterministic and Synchronous Multi-channel Extension (DSME)

This mode is intended to support industrial and commercial applications with strict requirements for punctuality and reliability. In DSME the time is divided into periods with contention (CAP) and without contention (CFP), as shown in Figure 11. The first timeslot (slot 0) is used to transmit EB, followed by the CAP contention period, which starts after EB and ends before slot 9. The other seven slots belong to the CFP contention free period, and each timeslot in the CFP represents a dedicated DSME-GTS (Guaranteed Time Slots), used when there is traffic whose latency should be predictable.

Throughout the CAP, frames are sent using only one channel and the nodes must be activated (connected). To use more GTSs and thus save energy, DSME provides a mechanism called CAP Reduction. When enabled, only the first superframe of each multi-superframe presents the CAP, while in the other superframes, the CAP is omitted and the CFP has 15 DSME-GTS.

In DSME, coordinators transmit EB periodically to keep nodes in sync and allow other nodes to join the network. The time interval between two signaling frames is called Beacon Interval (BI), which is composed of several superframes, that have no inactive periods, and are divided into 16 equally spaced slots. Within BI, it is possible to define repeated superframes cycles called multi-superframes.

During CAP, nodes use a CSMA/CA algorithm to access the channel and, in CFP, nodes use GTS. The difference from the IEEE 802.15.4 standard is that DSME extends the number of GTS timeslots and uses multiple channels. In addition, it can accommodate periodic and event-oriented traffic, adopting a versatile structure with multiple

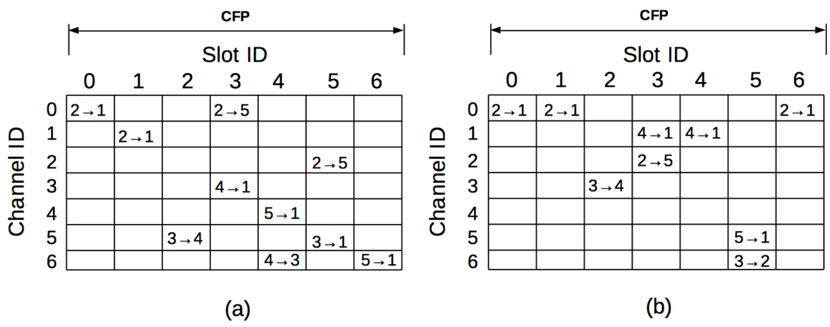

Figure 12: Channel diversity mechanisms defined for DSME networks.

superframes.

The beacons scheduling and the allocation of timeslots are performed in a distributed manner, without depending on a central device. Links between two nodes can be dedicated, so neighboring nodes can communicate point-to-point. As each pair of nodes can allocate and deallocate GTS timeslots, DSME is able to adapt to variations in traffic generated and changes in network topology over time. DSME also supports the option to confirm receipt of group packets (Group ACK - GACK) which allows the ACK of multiple data frames to be combined into a single ACK frame to improve energy efficiency.

Concerning GACK, it is used when nodes need to send periodic traffic to the coordinator. In this case, the coordinator uses a single DSME-GTS to gather all the ACK of data received in the previous DSME-GTS in just one frame. This mechanism has two advantages, energy efficiency and reduced latency, since the retransmissions are made within the same multi-superframe.

During the same GTS, multiple streams can be accommodated using different channels. Each superframe within a multi-superframe has a different ID, and each DSME-GTS has an ID according to its position within the CFP.

DSME mode defines two types of channel diversity: frequency hopping and channel adaptation. Figure 12 shows two examples of scheduling for the CFP period, using frequency hopping in (a) and channel adaptation in (b). In the first, nodes receive packets on different channels, depending on the displacement of the node's channel (channel offset), the slot ID, the superframe ID and the sequence number of the EB sent by the coordinator. For example, in Figure 12 (a), Node 1 receives a packet using Channel 0 in the first timeslot, using Channel 1 in the second timeslot, and so on. Nodes that receive packets within the same superframe must have different channel deviations to avoid collisions.

When using channel adaptation as a channel diversity technique, a pair of nodes com-

Parameters	Valor
SMS	2^{MO-SO} superframes
MSBI	2^{BO-MO} multi-superframes
MSD	aBaseSuperframeDuration $\times 2^{MO}$ symbols

Table 5: MAC parameters with multi-superframe.

municate over the same channel as long as it is good enough in terms of the signal-to-noise ratio. To use this mechanism, it is necessary to continuously evaluate the quality of the links.

In addition to both mechanisms, a new hybrid approach was developed and simulated in [D. Gomes et al. 2017], which uses frequency hopping and channel adaptation. The simulated results showed that the hybrid method surpassed the other two approaches for the studied scenario. The experiments also showed that the use of channel adaptation is better than the frequency hopping for transmitting packets in unicast mode. For packets transmitted in broadcast by the coordinator, the use of frequency hopping is a good alternative to deal with spatial variations in the quality of the channels.

The time frame in DSME is almost the same as in IEEE 802.15.4, with *macBeaconOrder* (BO) and *macSuperFrameOrder* (SO) to achieve low power consumption. The first specifies the period during which the coordinator can send the beacons. The second specifies the duration of the active portion plus the beacon [Khanafer, Guennoun and Mouftah 2014, Queiroz, Gomes and Benavente-Peces 2017]. In addition, DSME includes the *macMultisyperframeOrder* (MO) parameter, which defines the duration of multi-superframes, in which $0 \leq SO \leq MO \leq BO \leq 14$. In Table 5, SMS represents the number of superframes in a multi-superframe, MSBI means the number of multi-superframes in a BI, and the MSD is the duration of multi-superframes.

Each EB has a special field called *DSME PAN Descriptor Information Element*, which contains the information from the superframe, such as BO, SO and MO, and other options such as GACK, *CAP Reduction*, among others. The value of the SO influences the duration of the timeslot and the MO determines the number of DSME-GTS available in each multi-superframe, also influencing the latency.

DSME transmits a command called *DSME-Beacon Allocation Notification* during CAP to inform its neighbors about the allocation of timeslots. In the event of an EB collision, the coordinator uses the *Deferred Beacon* option to reduce the likelihood of a collision. When this option is enabled, the coordinator evaluates the channel using the

Clear Channel Assesment (CCA) before sending the EB. To prevent the same EB slot from being assigned to multiple nodes, a *DSME-Beacon Collision Notification* is sent to the other node to choose a different EB slot.

The experiments described in [Jeong and Lee 2012], [Lee and Jeong 2012] evaluated the performance of the DSME mode compared to the mode using IEEE 802.15.4 beacons. The experiments found that, in some scenarios, the transfer rate of the IEEE 802.15.4e DSME network can be 12 times higher than the IEEE 802.15.4 network with enabled beacons, and with less energy consumption, due to the use of a network access based on TDMA. In the experiments, the frequency hopping was used, but no dynamic denylist management was employed.

In [Sahoo, Pattanaik and Wu 2017] a new MAC protocol for networks with star topology based on the DSME mode is proposed with the objective of reducing the network discovery time and optimizing the bandwidth usage. To reduce discovery time, it has been proposed to use additional beacons streams on different channels. When a node tries to enter the network, a random channel is chosen and the node waits for a transmission of beacon on that channel. However, if the selected channel has problems of deep fading for the link between the new node and the coordinator, the access time may be very long.

This problem was not considered in the solution proposed in the article. To optimize bandwidth usage and reduce delay, GACK was employed using EB, and nodes use contention access periods to perform retransmissions instead of using dedicated timeslots for retransmissions, as defined in the original DSME algorithm [De Guglielmo, Brienza and Anastasi 2016]. A limitation of the work described in [Sahoo, Pattanaik and Wu 2017] is that the DSME channel diversity mechanisms have not been evaluated, which is a very important aspect to be considered in industrial applications.

3.3 Low Latency Deterministic Network (LLDN)

Few studies have addressed the LLDN to date. This mode is intended for applications such as factory automation and robot control, which require very low latency and determinism.

Communications in LLDN mode occur according to the superframe structure. Each superframe starts with a beacon packet broadcasted by the coordinator, and is followed by up to two timeslots used for management (optional). After that, several dedicated timeslots are defined for the transmission of the final nodes. When only one node is associated with a timeslot, it is not necessary to use an address, since the node can be identified by the ID of the timeslot, which is already known by the coordinator. Therefore, it is possible to transmit smaller packets and reduce network latency.

Unlike DSME, LLDN does not implement EB and there is no mechanism with multiple channels, and similar to DSME, it implements GACK. The access to the channel is achieved through dedicated timeslots through a version of CSMA/CA in shared timeslots. Dedicated timeslots are accessed using a TDMA approach, and the timeslots are smaller than those used in DSME.

Unlike DSME, LLDN was designed only for star topologies, in which several nodes must periodically send data to a central sink node. As communications take place on the basis of TDMA, the transmission between the final nodes and the network coordinator (sink node) is free of collision, so that it is possible to guarantee good quality in communications. However, problems related to shadowing, multipath, interference or asymmetry can degrade network performance. In addition, due to the use of a star topology, it can be difficult to accommodate a network with many nodes.

A possible solution to this problem is to use a coordinating node with several transceivers and divide the network into groups that use a common channel. Strategies for estimating link quality and channel allocation can be used to improve the quality of such networks.

An LLDN implementation described in [Anwar, Xia and Zhan 2015] uses IEEE 802.15.4 compatible radios, and aims to verify the feasibility of implementing applications that require intervals of up to 10 ms between the receipt of two new values from a sensor. In a scenario in which the sensor nodes send two bytes of payload, it was possible to accommodate eight final nodes, with a maximum delay between the reception of two packets of 9.996 ms. In a scenario where the sensor nodes send four bytes of payload, it was possible to accommodate seven final nodes, with a maximum delay of 9,536 ms.

A limitation of the analysis performed in [Anwar, Xia and Zhan 2015] is that the experiments were carried out in a controlled environment, without external interference, and with the nodes positioned very close to each other. Thus, the package delivery rate was 100% over the course of the experiment. In a real industrial environment, it may be practically impossible to achieve a 100% packet delivery rate, in which case, latency values may be worse in practice than shown in the article.

Some authors proposed channel diversity and multichannel transmission techniques for LLDN networks. In [Patti, Alderisi and Bello 2014], a multichannel protocol based on LLDN, called MC-LLDN, was proposed. MC-LLDN aims to increase the scalability of LLDN networks through the use of a tree topology, and data aggregation.

In this way, it is possible to accommodate simultaneous transmissions on the network on different channels. With the data aggregation on the routers, less timeslots need to be dedicated for forwarding packets to the coordinator, which causes reduction in latency. The limitation of the MC-LLDN is that the channels are statically allocated to the subnets, which makes the protocol unable to deal with the variations that may occur in the quality of the channels over time.

The protocol proposed in [Patti and Lo Bello 2016] is an evolution of MC-LLDN, called PriMuLa, which incorporates the adaptive selection of channels for the subnets. A limitation of PriMuLa, due to the characteristics of LLDN networks, is that the same channel is allocated to all nodes on the same subnet. However, spatial variations in channel quality can occur, as well as asymmetry problems. In addition, to perform the dynamic selection of channels, a mechanism for estimating the quality must be used in real time and does not cause an overload on the sensor nodes. The simulations to evaluate PriMuLa were performed considering only shadowing problems, through the use of realistic parameters for the log-normal shadowing model. However, other problems that can affect the quality of the links, such as the non-stationary behavior of the wireless channel in industrial environments, were not considered.

3.4 Time Slotted Channel Hopping (TSCH)

The TSCH mode was developed for process automation applications, uses EB, works in star, tree and mesh topologies, and employs a frequency hopping mechanism. Unlike DSME, which uses a periodic multi-superframe structure, TSCH uses periodic slotframes, dedicated timeslots instead of timeslots during CFP, and CSMA/CA with shared timeslots instead of contention-based channel access (during CAP). In addition, TSCH does not implement GACK.

In TSCH, each node can obtain synchronization information, frequency hopping sequence, and timeslots and slotframes through the EB. The slotframe contains timeslots based on contention or without contention. The assignment of timeslots to devices within the slotframe can initially be communicated through beacon, but generally the higher network layers configure them when the device joins the network [Alderisi et al. 2015].

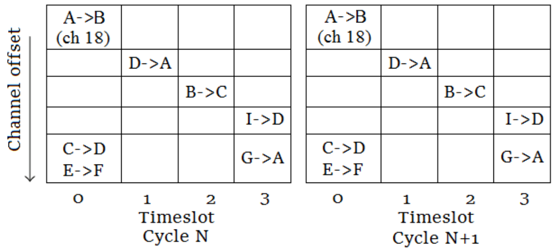

Figure 13: Example of TSCH slotframe.

Figure 13 shows an example of link schedule, in which there are four timeslots, five channel offsets and eight transmissions, thanks to the multi-channel approach of TSCH. In the example, there are five dedicated links (A $\rightarrow$ B, D $\rightarrow$ A, B $\rightarrow$ C, I $\rightarrow$ D, and G $\rightarrow$ A) and one shared link (C $\rightarrow$ D and E $\rightarrow$ F).

Since all devices share common time and channel information, devices can hop over the entire channel space to minimize the negative effects of fading and interference over multiple paths, and do so in a divided way in timeslots to avoid collisions, minimizing the need for retransmissions.

Through time synchronization and frequency hopping, TSCH allows for high reliability while keeping execution cycles (duty cycles) very low, and this ensures energy efficiency. The hopping sequence is defined by an ID, the length of the sequence and an ordered list of channels. A link between devices is identified by timeslot and channel offset. Shared links are intentionally assigned to multiple devices for transmission. This can lead to collisions and result in transmission failure detected by an absent ACK, but a retransmission backoff algorithm (CSMA/CA algorithm) is implemented for shared links, to reduce the likelihood of repeated collisions.

There are differences between the CSMA/CA of IEEE 802.15.4 and of TSCH [De Guglielmo, Brienza and Anastasi 2016]. The first difference is that in TSCH, in addition to being optional, CCA is not used to avoid collisions, but to transmit a packet if there is strong external interference. Another difference is related to the parameter *macMaxCSMABackoffs*, which is not used in TSCH. The packet is discarded only if it reaches the maximum number of retransmissions (*macMaxFrameRetries*). Regarding the backoff mechanism, it is activated only after a collision. In TSCH, the duration of the backoff unit, instead of $320\mu s$ from IEEE 802.15.4, corresponds to a shared timeslot.

Each timeslot on the TSCH network has an associated channel offset, which is converted to a frequency in a pseudo-random hopping method. A link between two nodes can be represented by a pair between timeslot in slotframe (n), and the channel offset used by the nodes in that time interval, defined as [*n, channelOffset*]. The frequency f to be used to transmit in timeslot n of the slotframe can be calculated as follows:

$$f = FHS[(ASN + CO) \bmod FHS_L], \tag{3.1}$$

in which the Absolute Slot Number (ASN) represents the total number of timeslots incremented in each timeslot since the beginning of the network, % *CO* is the channel offset variable used by the nodes at a timeslot n, defined as [*n, channelOffset*], mod is the modulo operator, FHS_L is the number of available channels, without using denylist in the calculation. Thereafter the function FHS[.] generates a number between 0 and 15, with the value 0 being represented by channel 11 (2.405 GHz) and 15 being represented by channel 26 (2.480 GHz), with a difference of 5 MHz between adjacent channels.

As in WirelessHART, in TSCH each timeslot lasts 10 ms, and allows the transmission of a data packet and the corresponding ACK confirmation packet. If a given transmitter does not receive the ACK packet within the same timeslot, it can relay that packet in the next timeslot allocated to it, even during the same slotframe, or in the next slotframe, depending on the scheduling policy assigned to it.

In [Alderisi et al. 2015], a comparison between DSME and TSCH is described in process automation scenarios. Simulations were performed to check the delay, reliability and scalability for each mode. TSCH showed better results for small networks, with up to 30 nodes. For larger networks, with more than 30 nodes, DSME presented better results. The simulations used realistic parameters for log-normal shadowing, but the effect of fading and the non-stationary characteristics of the wireless channel were not considered. In addition, only the DSME frequency hopping mechanism was analyzed.

In [Juc et al. 2016], another comparison between DSME and TSCH is described, in terms of energy consumption and performance. In the considered scenarios, the energy consumption and performance of DSME were slightly better than TSCH. In applications that send less data, TSCH wastes bandwidth, due to the fixed size of timeslots. In the experiments, only the frequency hopping was considered.

3.5 Channel Denylists

It is a challenging task to configure the channel denylists and at the same time define an appropriate scheduling of transmissions so that two or more neighboring nodes do not use the same channels at the same time. Although it presents an improvement in the general performance, the use of denylists decreases the capacity and diversity of the network, since the same traffic must be sent through a smaller number of channels.

There are two approaches to creating denylists: global and local. In the first, all nodes use the same list, but this solution tends to perform below ideal, as the channels have different quality in the different network links, even if the nodes are close to each other [Gomes et al. 2017]. In the local method, each link uses different sets of channels [Kotsiou et al. 2017], and the more dynamic the environment, the more varied the denylist is. Despite surpassing the global list in terms of frequency selective fading, its management is much more complex and is subject to problems of internal interference and collisions.

Parallel transmission on the same channel results in several nodes transmitting in broadcast at the same time, and if the environment is very unstable and the denylist is modified very often, the local denylist may not be useful due to the excess of signaling frames required for resynchronization.

When it comes to performance optimization, the denylist, routing and time scheduling should be defined together, however, joining the three methods brings high complexity to the algorithm [Gunatilaka, Sha and Lu 2017], given the difficulty in managing heterogeneous denylists. Most approaches first propose the construction of denylists and then modify the scheduling as needed [Kotsiou et al. 2019].

To create a denylist, it is necessary to analyze the channels previously and, for this purpose the use of a LQE is required, which is a dedicated device or an integrated sensor function. To guarantee a high performance of the network, the estimation process must be done continuously.

The importance of the correct management of the channel denylist has been neglected by most studies when working with TSCH, which focus more on scheduling and frequency hopping in a blind way, without considering the quality of the channels.

Using Equation 3.1 to define the channels the nodes will hop to, Table 6 shows some examples of the problems related to simultaneous transmissions with the same channel

ASN	Channel Offset	Available Channels	Equation	Channel ID (f)	Links	Configuration
2	5	16	(ASN (2) + Offset (5)) % 16	7	A -> B D -> F	Same Channel Offset
3	5	16	(ASN (3) + Offset (5)) % 16	8	E -> G H -> I	
2	5	16	(ASN (2) + Offset (5)) % 16	7	A -> B	Different Channel Offsets
2	6	16	(ASN (2) + Offset (6)) % 16	8	D -> F	
3	5	16	(ASN (3) + Offset (5)) % 16	8	E -> G	
3	8	16	(ASN (3) + Offset (8)) % 16	11	H -> I	

Table 6: Channel Offset problem with global denylists related to Figure 14.

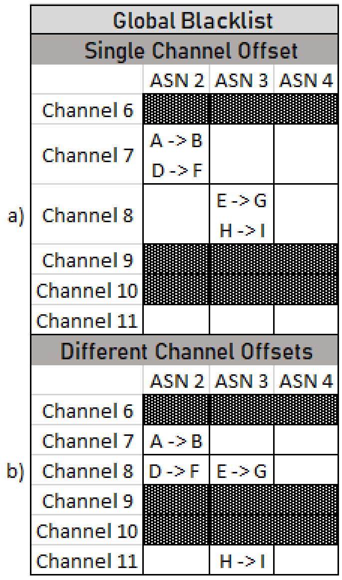

Figure 14: Problem of global denylists when assigning channel offsets.

offset, and Figure 14 shows the use of a global denylisting approach.

In the first row of Table 6 there are two links (A -> B and D -> F) transmitting at the same time, i.e., in the same timeslot (ASN 2). Using the same channel offset for both, the channel resulting from the equation is 7, as depicted in Figure 14 (a). If there is no way to prioritize which device is preferred in transmission, there will be collisions in that communication. The same occurs on the E-> G and H-> I links with the same channel offset (5). By assigning different offsets (5,6 and 5,8), also depicted in Figure 14 (b), the nodes will hop to different channels. In the global denylist approach of Figure 14, each link in the network has the same denylist (channels 6, 9 and 10), and as already discussed, it is inefficient to deal with frequency selective fading.

Initially, channel offset 5 was assigned to links A-> B and E-> G, and channel offset 6 assigned to links D-> F and H-> I. However, the resulting channel of H->I link is denylisted (channel 9), and then the array of channel offsets is incremented until it reaches a allowlisted channel. In Chapter 4 the channel offset array is explained in details.

The use of different channel offsets can mitigate the collision problem, however if the end nodes are close together, it is very likely that the transmission receives internal interference if the chosen channels are adjacent or spectrally close to each other. To circumvent this problem, it is necessary to choose spectrally distant channels.

Figure 15 shows an improvement of the approach with local denylists. Each link has its own denylist, and the likelihood of occurring internal interference is slightly lower compared to the global denylist, however it is still necessary to define spectrally distant channels through the proper management of the different channel offsets.

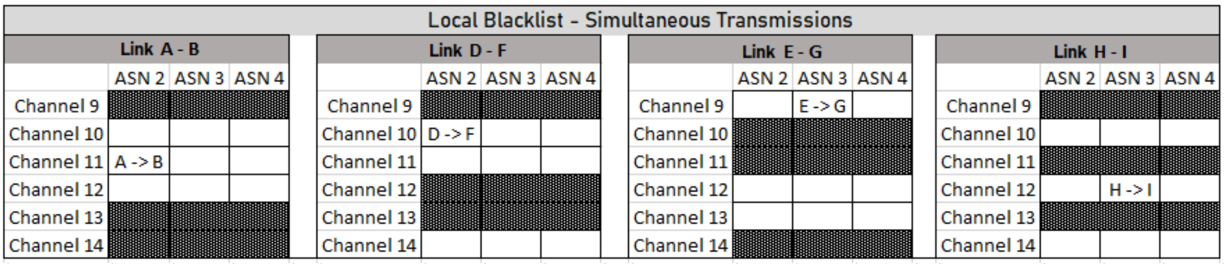

Figure 15: Local denylist with simultaneous transmissions.

3.5.1 Related Papers of Channel Denylisting

As mentioned previously, the concept of denylist is discussed more often in other related works, as it is the most researched in the literature. Few studies have addressed extra lists to increase network performance.

In [Sha, Hackmann and Lu 2011], the channels are denylisted when they present quality below a threshold related to the Expected Transmission Count (ETX) quality metric. One disadvantage of the algorithm is that it is more suitable for residential or office environments, in which the communication between the nodes is disturbed by interference produced by Wi-Fi networks. In that environment, according to the authors, only a few frequency hops are needed to maintain good quality in the communication, around 95% of PDR for up to 36 hops per day. In the article they also discuss the fact that adjacent channels generally exhibit similar behavior.

The problem of simultaneous transmissions on adjacent channels is very relative, because when the devices are very far apart, the interference between them can be negligible. It occurs when the sensors are close to each other, which can generate the so-called near-far effect. For the sake of simplification, most studies do not consider interference in adjacent channels.

The routing layer with channel denylists was studied in [Gnawali et al.]. The authors argue that denylisting policy can dramatically decrease routing options, limiting the efficiency of the algorithm if no limit is established. They demonstrate that in a network with multiple hops, the ETX metric with retransmissions provides better overall performance, while the use of denylists outperforms networks with high density.

Regarding routing, some authors discuss whether packet losses caused by blocking links and shadowing should be resolved by redirecting to alternative routes [Rashvand and Abedi 2017]. However, the alternative route does not solve the problem when there is interference and the only solution is the denylist approach.

Tang et al. [Tang et al. 2011] propose a protocol called EM-MAC, considered as a semi-dynamic multichannel protocol, which performs a channel denylist using CCA. Just before the transmission of packets, a channel evaluation is made by the CCA method to detect whether a channel is free from interference. Transmission begins only if the receiver allows it, and channels are denylisted if CCA fails or the packet delivery fails three times. The transmitting nodes are then informed of the denylists updated by the corresponding receiving nodes. After a pre-defined period, the channel is then denylisted. In addition, the denylist is periodically deleted and the nodes need to make new estimation.

In [Du and Roussos 2013] it is presented the Adaptive TSCH (A-TSCH), which to make decisions on the denylist, uses only a signal strength estimate based on hardware called Energy Detection (ED). The transmitting nodes are aware of the neighbors' denylist, and the transmitter and receiver use the same hopping sequence to communicate, inserting the list information into the transmitted packets. It reserves two timeslots in each slotframe to measure the noise level, and these samples are used to define a channel quality factor. Periodically, a fixed-length list of the best channels is chosen. This approach, however, does not take local interference into account nor does it use the PDR as an associated metric.

In addition to interference, other aspects that can affect the link quality are not considered in A-TSCH, such as shadowing and fading. Finally, the monitoring of the quality is performed by all nodes and using timeslots that could be used for data transmission, which generates a high overhead and increases latency.

Comparing with EM-MAC, A-TSCH performs better, as it regularly monitors the spectral condition and proactively protects the network against possible adverse factors.

EM-MAC awaits CCA failure or unsuccessful package delivery to work against the negative effects. Despite the advantages, A-TSCH uses extra timeslots for channel sampling and traffic control, which reduces network capacity and leads to wasted energy for the node to remain idle.

A spectrum detection technique is used during some dedicated timeslots to identify which channels are assigned to the denylist [Chiti, Fantacci and Tani 2015]. The work on [Mathew and Manuel 2015] also uses some dedicated timeslots with a spectrum monitoring technique for denylisting channels. In both, specific cells are needed where transmission is not allowed. As discussed earlier, using extra timeslots wastes bandwidth and energy.

In [Tavakoli et al. 2018] a dynamic allowlist method called Enhanced TSCH (ETSCH) is proposed, which uses a non-intrusive technique for estimating channel quality called NICE. ETSCH uses the frequency spectrum ED method to measure channel quality at least twice per timeslot. This makes ETSCH perform better than A-TSCH on very dynamic networks. It improves the accuracy of the A-TSCH assessment due to the more frequent RSSI sampling, and establishes a secondary list of the best channels through which EBs are transmitted. The use of a secondary list reduces the likelihood of losing EBs. A-TSCH uses all sixteen channels to transmit EB periodically, which can lead to losses of those EB and, consequently, loss of synchronization between the nodes.

Monitoring the PDR from broadcast packets such as EB can help to deduce the quality of the channel, however, depending on the number of nodes, it can cause long delays, because when some more distant nodes receive information from beacon, the local quality has probably already changed in very dynamic environments.

The use of dedicated EBs to scan for bad channels might not be recommended, as it wastes energy and adds unnecessary extra traffic, since the analysis must be done for each channel in the denylist of each link. One solution to mitigate this problem is to scan a bad channel less frequently than a good channel.

In [Gursu et al. 2016] an experiment is carried out inside the airplane cabin with Wi-Fi interference. Using the 16 channels, the network presented a packet error rate (PER) of 35%, reaching a performance of 5% PER with a single channel. This means that working with fewer channels reduces interference, however, the interfering channels are not completely denylisted in the solution. In addition, Wi-Fi networks hardly change the channel on which they operate, which explains the significant improvement in performance with fewer channels.

ETSCH [Tavakoli et al. 2018] explores the idle time at the beginning of each cell to detect external interference. As with A-TSCH, ETSCH only takes into account interference problems. Unlike A-TSCH, idle spaces within timeslots are used to estimate quality, avoiding the use of dedicated timeslots for this purpose. Like A-TSCH, they do not detect frequency selective fading on certain links, do not consider simultaneous transmissions, and do not use PDR as a quality metric. In addition, an overhead is imposed on the end nodes, as they need to identify sources of interference hidden from the coordinating node.

Blocking channels based solely on CCA does not appear to cause a significant increase in network performance, since the CCA prevents only interference problems. CCA failures are likely to be caused by interference problems, and when consecutive transmissions fail, they are likely to be caused by deep fading. Other metrics, such as RSSI and PDR, in conjunction with CCA, could help to make a better estimate.

The work in [Hänninen et al. 2011] propose denylisting channels if they have an average RSSI value below a threshold. However, studies show that RSSI calculates imprecisely the quality of links [Gomes et al. 2017].

RSSI and LQI work as channel quality indicators, as the level of the received signal is correlated with the Bit Error Rate (BER). However, as they can only be measured in correctly decoded packages, they fail to estimate quality when they are not associated with the PDR for each channel. Under interference, packets may be corrupted, making estimation difficult.

The distributed denylist protocol called Multi-Armed Bandit Optimization (MABO-TSCH) is presented in [Gomes, Watteyne and Krishnamachari 2017]. This approach covers three algorithms: the first assigns several channel offsets in the same timeslot to avoid interference, in which the nodes choose an offset that does not provide a physical frequency in the denylist; the second proposes a local denylist negotiation technique, in which data and recognition packages (ACK) are used to disseminate local lists to neighbors for negotiation purposes; and the third algorithm calculates the channel based on the Stochastic Multi-Armed Bandits MAB problem.

This method provides near-optimal results without the need for a learning phase. A pair of nodes locally decide the channels to be denylisted and, to maintain a consistent list for the link, they insert it into the ACK packet, combined with a sequence number.

The MABO-TSCH protocol is suitable for large-scale networks in which each node has few neighbors, reducing the need for much communication between them. However, it is inefficient on small, dense networks. The generation of the radio channel is an iterative process in which a offset is used for time. If no channel in the allowlist is generated with the first chosen offset, the second offset will be used, and so on. If everyone fails to generate a channel in the allowlist, the node will postpone its transmission in the current timeslot, causing delays and extra power consumption, since the receiver turns on the radio even though no packets are sent.

In [Dakdouk et al. 2018] an approach similar to MABO is proposed. Each node receives a set of channel offsets and, after that, uses the MAB algorithm to select a channel among those allowed based on a reward. The approach presents an improvement of MABO and proposes to dynamically allocate the offset in each timeslot according to the number of simultaneous transmissions to avoid collisions.

The authors in [Kotsiou et al. 2019] propose to group radio links so that all links in a group share the same allowlist of channels, and different groups have different allowlists. The idea is to avoid a global allowlist, which decreases network performance, and a local allowlist, which can cause collisions, as explained in this section. The solution forces all links in the same timeslot to use the same denylist, which is a sub-optimal method, although it avoids collisions.

This method showed better results than approaches without a denylist, with a global denylist, with a common denylist by timeslot and with the MABO-TSCH algorithm. However, interference problems can still occur due to spectrally close channels.

A channel selection chosen by the transmitter based on ACK packets and CCA failures is implemented in [Peishuo Li et al. 2015], in which the channel list is included in the data packets sent to the coordinator. The coordinator then broadcasts the new list to avoid synchronization problems. The selection is associated with a reward measure called Gittins Index, and this index is increased when the packet is successfully transmitted, and set to zero when failures in the busy channel or consecutive transmission failures occur.

A disadvantage of this approach is that packet transmission status is only available for transmissions that require the ACK acknowledgment packet, while most real-time and multicast communications do not use ACK packets. End nodes generally have limited resources and should not make quality estimates. In addition, according to [Tavakoli et al. 2018], it is local interference on the receiver side that affects communications.

Elsts et al. [Elsts et al. 2017] combines central allowlist and distributed denylist. The coordinator generates samples of RSSI and provides a allowlist. EBs are used to distribute the allowlist, and each node generates a denylist based on its PDR. The node uses the channels that are in the central allowlist, but not in the local denylist. A disadvantage of this method is that the RSSI procedure on different channels is not specified. In addition, it makes the number of available channel offsets unpredictable.

The LaBeL algorithm [Kotsiou et al. 2017] executes a allowlist and maintains at least three channels in it on each link. It continuously estimates the performance of PDR for all channels, including those on the denylist, and modifies the pseudo-random hop sequence to send data packets across denylisted channels to update the PDR value.

In LaBeL, the Window Mean Exponential Weighted Average (WMEWMA) estimator is used to measure the average PDR for each channel and the neighbor. Based on this, the best channels are selected to determine the threshold for detecting bad channels. A pair of transmitter and receiver autonomously decides which allowlist to use at the beginning of timeslot. In fact, it recalculates the channel using Equation 3.1 with a pseudo-random variable derived from the transmitter ID, until a channel in the allowlist is selected. The pseudo-random approach can create collisions between links with different channel offsets, as discussed in this work, and can make the network non-deterministic and less suitable for critical operations.

In [Banik et al. 2018], instead of estimating the link quality using indicators such as RSSI, it analyzes the statistics collected through a cloud application, which decides which channel needs to be denylisted. It uses an additional timeslot to classify denylisted channels using a short dummy packet. This procedure is repeated several times for each channel of each link. As discussed in this paper, the use of extra timeslots generates high overhead and increases latency.

In [Zorbas et al. 2018] the LOST algorithm is presented and is based on the information collected by its neighbors just a hop away. It multiplexes the different transmissions between different channels, appropriately allocating channel offsets. A local denylist method is employed on the scheduler to avoid using low quality channels.

In [Zorbas, Papadopoulos and Douligeris 2018] a global denylist approach is presented with only one channel offset. The authors point out that the local denylist method suffers from limited offsets; therefore, it uses a global denylisted algorithm to reduce interference. Each node maintains a temporary denylist associated with a duration. When the time expires, the channel is permanently denylisted. Whenever a channel is moved to this list, this information must be distributed to the entire network, encapsulated in data packets or

Protocol	Based TSCH	Method BL/WL	Resets BL/WL	Extra Timeslot	Channel Offsets	Propagation BL/WL	Approach BL/WL	Postpones Transm.	Interf.	Multipath Fading.
ARCH	No	ETX	No	No	No	ACK or Broadcast	Local	No	Yes	No
EM-MAC	No	CCA	Yes	No	No	Beacons	Local	No	Yes	No
A-TSCH	Yes	ED	No	Yes	Single	Beacons	Local	No	Yes	No
ETSCH	Yes	ED	No	No	Single	Beacons	OM	No	Yes	No
MABO-TSCH	Yes	MAB	No	No	Multiple	ACK and Data	Local	Yes	Yes	Yes
[Kotsiou et al. 2019]	Yes	TRP	No	No	Multiple	OM	Hybrid	No	Yes	No
[Peishuo Li et al. 2015]	Yes	ACK and CCA	No	No	OM	Data and Broadcast	OM	No	Yes	Yes
[Elsts et al. 2017]	Yes	RSSI	No	No	OM	Beacons	Hybrid	No	Yes	Yes
LaBeL	Yes	TRP	No	No	Multiple	Beacons	Local	No	Yes	No
LOST	Yes	TRP	No	Yes	Multiple	Beacons	Local	No	Yes	No
[Zorbas, Papadopoulos and Douligeris 2018]	Yes	TRP	No	No	Single	ACK and Data	Global	No	Yes	No

Table 7: Main works related to the thesis theme.

ACK, so that all nodes maintain the same denylist for future calculations of the channel. Compared to MABO, this method showed a higher PDR and reduced delay.

The channel denylists in star and tree topology using the TSCH protocol was analyzed in [Queiroz et al. 2018]. In tree topology, the network was divided into clusters, and in each cluster there is a CH that works as LQE, network coordinator and router, and forwards packets from the end nodes to the sink node. The quality estimation is carried out locally, and the results showed that it is important to implement the denylist approach, however, there must be a limit on the number of blocked channels. This does not happen in the star topology, in which the more channels are denylisted, the higher the PDR.

The ideal size of the denylist depends on many factors, such as the dynamics of the environment, the size of the network, the topology, the number of nodes with simultaneous transmissions, and the study of the ideal number must be done carefully before the deployment of the network. In a star topology, for example, in which there is no simultaneous transmission and each node has a transmission time interval, it is possible to use a denylist of 15 channels, with only one available for transmission (the best of all, according to the estimator).

Table 7 presents a summary of the main works related to the thesis theme. The columns deal with whether the protocol is based on TSCH, what is the method for generating the denylist, whether the denylist is removed entirely during the progress of the network, whether extra timeslots are used, whether channel offsets are used, how the list is disseminated throughout the network, if a global or local approach is used (OM means that the work did not make it clear which one was used or was omitted), if it postpones transmission in case of failure, if the work had as focus on interference and multipath fading.

3.6 Final Considerations

In this chapter, an overview of the IEEE 802.15.4e standard and its respective operation modes were presented. Among them, the most studied in literature is TSCH, and several solutions have been proposed for this protocol, the vast majority of which are related to time scheduling. There are still several open issues, such as the methods of channel hopping adaptation to use the most reliable channels throuth channels lists defined by link quality estimators. When working together, these methods increase significantly the performance of the network in harsh environments.

As discussed, several studies have been developed with the aim of mitigating the negative effects of the wireless media on sensor communications. Some of them use metrics that estimate the quality for generating the denylist, such as ETX, RSSI, ED or even the number of failed transmissions. Others use the MAB problem and/or the CCA method to detect whether a channel is free from interference, or use only the PDR.

Although one metric may perform better than the other, they should work together, as there are factors that they alone cannot predict, such as shadowing, fading, non-stationary characteristics of channels in dynamic environments, as in industry, as well as differences in the characteristics of the different channels.

Some works define extra timeslots in the scheduling, or use EB to evaluate each channel, but before applying these methods, it is important to consider the delay it causes in the network, energy consumption and overload, especially in large networks and dynamic environments. Other works focus on studying how the denylist should be distributed over the network. Some of them use data packets and ACK, others use EBs.

Another open question is how to monitor the quality of channels that are already denylisted. Using past communication history from channels that were previously allowlisted and now denylisted may not work well in dynamic environments [Tavakoli et al. 2018]. Some works periodically remove the denylist, others select some channels from it and transmit some packets to estimate their quality, and then reinsert them in case of failure to receive packages.

Some studies focus on properly assigning channel offsets to mitigate interference along with the local denylist approach. In this case, TSCH defines the offset variable in Equation 3.1 so that the sensors avoid choosing the channels compromised by internal (neighbors) and external (Wi-Fi, for example) interference. Some works define a single offset, others define several offsets, and in the next section a solution called Adaptive-Blacklist

TSCH (AB-TSCH) is proposed to mitigate the collision and interference problem.

4 Adaptive-Blacklist TSCH

To deal with reliability problems in IWSN, mechanisms that allow the network to adapt itself to the variations that occur over time in the quality of the links need to be implemented. In this chapter, a new protocol for IWSN is proposed, called Adaptive-Blacklist TSCH (AB-TSCH), which uses both frequency hopping and channel denylisting, together with the LQE approach to mitigate the problems mentioned in the thesis.

4.1 Description of the AB-TSCH Protocol

AB-TSCH is a protocol for WSN focused on industrial environments, although it can be used in residential and office settings. It is based on the frequency hopping method established in the IEEE 802.15.4e TSCH protocol, and uses the idea of a local denylist similar to the ISA110.11a standard, but without postponing the transmission if it finds a denylisted channel.

The AB-TSCH protocol can be used in networks with tree or star topology. Figure 16 shows an example of tree topology studied in this work, which is more complex than the star topology due to the possibility of simultaneous transmissions, collisions and internal interference.

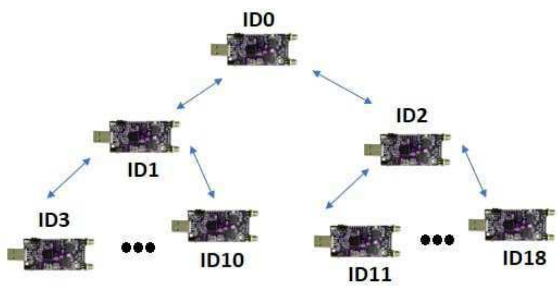

Figure 16: Nodes identified by their ID, in which ID0 is the sink node, ID1 and ID2 are cluster heads in tree topology, and the rest are end nodes.

In the configuration used to evaluate the protocol, 16 end nodes (ID3 to ID18) were

Number of Packets	Number of denylisted channels
200	1
400	2
600	3
800	4
1000	5
1200	6
1400	7
1600	8
1800	Denylist flushed

Table 8: Procedure for channel denylisting

deployed that only sense the environment and transmit the data. In star topology, the receiver is a sink node that works as network coordinator and LQE, and in tree topology two cluster heads (ID1 and ID2) are added, that also work as coordinators and LQEs, each one for their own subnetwork. They receive the data packets from the end nodes and forward them to the sink node (ID0).

The end nodes transmit only once per timeslot and all of them have the same priority, with one retransmission in the next slotframe, if necessary. As the time goes on, the links present different denylists, and the allowlisted channels are not necessarily be the best, but at least they are not the worst channels in performance and reliability.

The LQE function of ID0, ID1 and ID2 nodes assigns to their links the channels the network can use during the next period (cycle), i.e., channels that performed well during the previous cycle. The cycle is related to the number of packets received by the LQEs, as depicted in Table 8.

The network initially has no denylisted channels, i.e., all channels can be used. For each cycle of 200 received packets, the worst channel of that cycle is denylisted, totaling eight denylisted channels in the eighth cycles. After that, the denylist is flushed and the process is restarted.

Frequency hopping is used to transmit enhanced beacon packets (EB) from the network coordinators, and data packets from the end nodes. The idea of using frequency hopping is to deal with spatial variations in channel quality. In addition, the channel denylist is dynamically configured to avoid the use of channels with poor quality. The LQE uses the RSSI from data and ACK packets (i.e. from both sides of the communication), the number of duplicated packets and the number of retransmissions to build the channel denylist.

In AB-TSCH, the medium access is based on a structure called multislotframe, similar to the structures defined for the DSME and TSCH modes. Each multislotframe is formed by a set of slotframes, and each slotframe is composed by a set of timeslots, in which the first timeslot in each slotframe is used for the transmission of EBs and the others for data packets, as depicted in Figure 17.

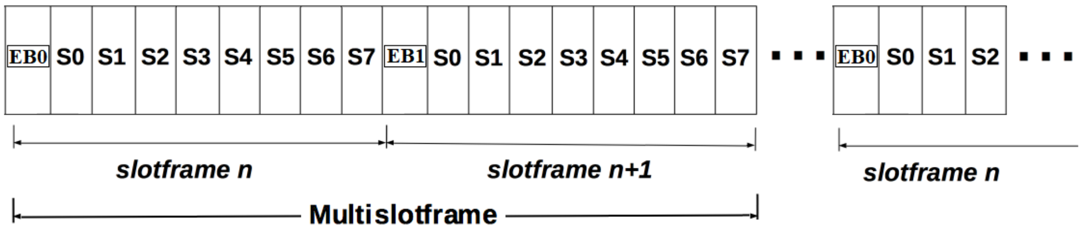

Figure 17: Multislotframe with several timeslots, including the EB timeslot at the beginning of each slotframe.

After receiving and processing the first EB, represented as $EB0$, to synchronize to the network, the nodes can receive but do not need to process the next EBs (EB1, EB2, and so on) in the same multislotframe to maintain the communication, unless they loose connection. With this approach, the network performance is less affected by overhead problems with receiving EBs, compared to other protocols, such as DSME and the EB-enabled mode of the IEEE 802.15.4 standard.

All slotframes within a multislotframe have the same structure, with the timeslots allocated to the same nodes, and the assignment of timeslots is based on the requirements of the application run by each end node.

Figure 18 presents a simple diagram of the protocol AB-TSCH. The first lane shows that the end nodes wait for EBs, and if many EBs are lost in sequence (this threshold can be configured), the end node enters a recovery state, in which it listens in each of the 16 channels during given slotframe periods, until a EB is received.

Once the node receives an EB, it retrieves the information about the network and waits its own timeslot to transmit the data packet. As mentioned before, if another EB is received in the meantime, the node skips it, or if the EB (connection) is lost, it returns to the waiting stage for the next EB.

When the node is allowed to transmit, it calculates the channel based on Equation 3.1 and considering the channel denylist until it finds an allowlisted channel and then transmits the packet. After the transmission, the node waits for an ACK packet from the receiver on the same timeslot, which lasts 10 ms (transmission of data packet and reception of ACK packet). If the ACK packet is received, and there is no timelost

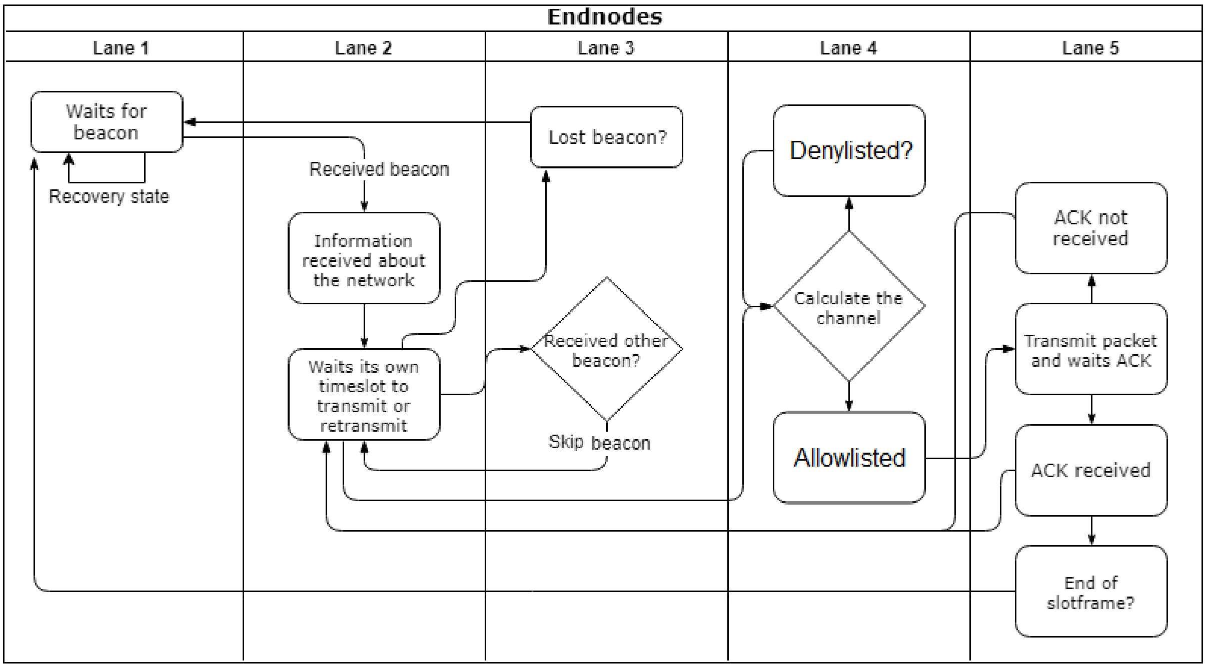

Figure 18: Diagram of the end nodes steps.

assigned to the node in the current slotframe, it waits for the next EB in the next slotframe. As mentioned, the next EB does not need to be processed, unless the network reaches the next multislotframe, i.e., the next $EB0$, or the node loses connection. If the ACK packet is not received, the node waits the next timeslot assigned to it to transmit the same packet.

The retransmission probably occurs due to a leaf-to-root communication failure. However, when the CH or sink node receive the transmitted packet but the ACK packet is not received by the transmitter, the latter retransmits the same packet in the next timeslot, and the CH or sink consider as a duplicated packet, i.e., there is a root-to-leaf communication problem.

The transmission of EBs from the network coordinators occurs simultaneously, using different channels. Although the list of channels used to transmit them is the same for all coordinators, each one uses a different offset. The sink node (ID0) transmits the first EB using the first channel on the list (EB_T), with the cluster heads ID1 and ID2 remain waiting to receive it (EB_R). After receiving, the cluster heads forward the EB using the second and third channels on the list (EB_T), respectively, in the next timeslot. This process is depicted in Figure 19. Before the networks starts, the cluster heads need to receive the EB from the sink node, and then forward it to the end nodes.

Every data and ACK packets transmitted has a field of denylisted channels that

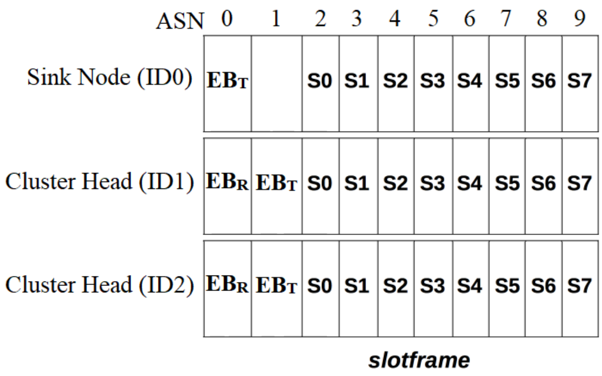

Figure 19: Slotframe with EBs in tree topology.

the transmitter and receiver are aware not to use. In this case, EB are used only for synchronization, not for broadcasting the denylist. This may be justified by the fact that by losing synchronization and then resynchronizing using the following EBs, depending on the size of the network, the denylist information entered in the EB might be outdated in very dynamic networks. To synchronize with the network, the node hops through all the channels, waiting for the EB.

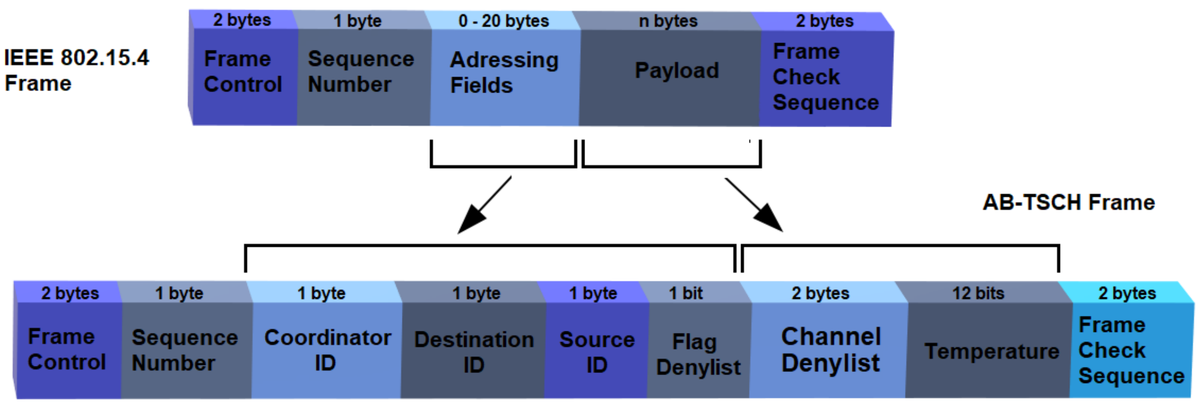

Figure 20: Data packet from IEEE 802.15.4 standard and AB-TSCH.

Figure 20 illustrates the structure of the data packet transmitted by the end nodes. The addressing fields are used to store the coordinator ID, destination ID, source ID, and the flag denylist, which is used to inform the receiver that the transmitter is aware about the updated denylist.

The payload consists of two fields, the first one representing the denylist with 16 channels (16 bits), and the second with 12 bits representing the temperature, in which 11 bits are set for the data and 1 bit for the operator (+ or -). For example: with channels 3, 9 and 15 denylisted, and the temperature monitored about -15° C, the payload is the same as $1000\ 0010\ 0000\ 1000_2$ for the channel denylist, $0111\ 1101\ 1111\ 0111_2$ for the channel allowlist (channels 0, 1, 2, 5, 6, 7, 8, 10, 11, 12, 13, 14 and 15), the operator 1_2 due to the negative value of the temperature, and 000 0000 1111_2 for the temperature.

The temperature value is not related to the developed protocol, however it illustrates that it is possible to insert data into the payload from the environmental sensing, in addition to what the protocol defines.

One open issue remains at the beginning of the network. Most studies do not mention how channel denylisting is done at this time, since the quality of all channels is unknown. In this protocol, at the beginning of the network 16-bit denylist field is set to zero, since there is no denylisted channel.

After the first analysis of the LQE, one channel is denylisted, and the receiver inserts into the ACK packet the information of the channel denylist after it received the data packet from the transmitter. Once the latter received the ACK packet, it updates its own denylist field with that channel, and transmits the new data packet with the updated information and the flag 1_2 activated. After the destination received this packet, it compares with the latest ACK packet to know if the transmitter is aware of the denylisted channel. If it matches, the receiver then sends the ACK packet to inform the transmitter that now they can communicate using the new channel denylist. This procedure is depicted in Figure 21.

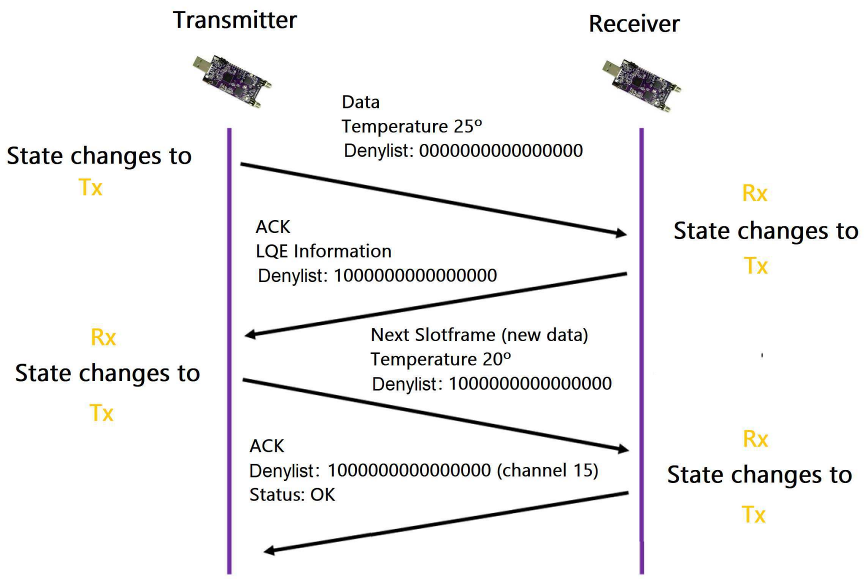

Figure 21: Procedure of transmitting the denylist information from the LQE nodes to the end nodes.

In the example of Figure 21, the receiver (LQE) informs the transmitter that channel 15 should be denylisted. The link only starts to use the updated denylist when the transmitter receives an OK message from the receiver, which is inside the ACK packet.

Another open question is how to monitor the quality of the already denylisted channels, since they could not be used by the nodes. Using the history of past communications of the previously allowlisted and now denylisted channels may not work well in dynamic environments [Tavakoli et al. 2018]. Some works flush the denylist periodically, others select some channels from the denylist and transmit some packets through to estimate their quality, and then denylist again in case of packet reception failure.

In the developed protocol, one channel is denylisted at a time (the one that presents the worst performance) until the algorithm reaches eight denylisted channels. After that, a new estimation is made, and the denylist is flushed. In very dynamic environments, the previously denylisted channel will probably not present the same quality and might not be denylisted again for some time. This approach allows to ensure diversity, while avoiding the lowest quality channels.

```
node[0].MAC.nodeIdSlot = "00 00 00 00 00 00 00 00 01 02 01 02 01 02"
node[1].MAC.isClusterHead = true
node[1].MAC.clusterID = 0
node[1].MAC.macBufferSize = 12
node[1].MAC.nodeIdSlot = "03 04 05 06 07 08 09 10 01 00 01 00 01 00"
node[2].MAC.isClusterHead = true
node[2].MAC.clusterID = 0
node[2].MAC.macBufferSize = 12
node[2].MAC.nodeIdSlot = "18 17 16 15 14 13 12 11 00 02 00 02 00 02"
```

Figure 22: Time schedule of the network coordinators.

Figure 22 shows the time schedule of the three coordinators. The sink node (node[0]) receives data from both cluster heads (node[1] and node[2]), and may turn off its transceiver for the first eight timeslots (parameter *nodeIdSlot* with value *00*). In the meantime, the cluster heads receive data packets from the end nodes to which they are directly connected. Once all packets from the end nodes are received by the cluster heads, the sink node turns on its transceiver to receive the respective packets.

```
int offset1[8] = {0,1,4,5,8,9,12,13};
int offset2[8] = {2,3,6,7,10,11,14,15};
```

Figure 23: Channel offsets of each cluster head, for the nodes to choose spectrally distant channels.

The multiple channel offset approach is used to increase the chances of generating a channel with good quality as well as to avoid frequency overlapping with neighbors that transmit on the same timeslot. However, there is a high probability of none of the assigned offsets generate an allowlisted channel, especially with large denylists. Since the denylist size depends on external factors, the key point in local denylisting is how many channel offsets should be available.

In this protocol, a hybrid approach is defined to assign channel offsets. A single one is set to the communication among the sink node and CHs, since they do not transmit at the same time (each one has its own timeslot), and eight static channel offsets for each link among the CHs and the end nodes, in a way that two parallel transmissions use different non-adjacent channels, as depicted in Figure 23.

The EB policy and the channels by which they are transmitted (all the 16 channels or through only the allowlisted) are also an open issue, and in this work the EBs are transmitted through all 16 channels, without the denylisting approach. This provides diversity, allows new nodes not aware of the denylisted channels to synchronize with the network, and it is not limited to a single channel that might be compromised by interference and multipath problems.

4.2 Simulation Results

A simulation model described in Section 2.3.3 was developed to simulate multi-channel protocols for IWSN in [Gomes et al. 2017]. The channel model captures the effects of fading, shadowing and non-stationary channel characteristics in industrial environments, as well as considering differences in the behavior and link asymmetry of the 16 channels. The model was integrated into Castalia and used in this work to evaluate the performance of the proposal in industry.

In the experiments, several comparisons were made involving star and tree topologies, with varying sizes of channel denylists, with and without collisions. The first experiments are depicted in Figure 24, and the results were published in [Queiroz et al. 2018].

In Figure 24 (a), PDR values are shown in the application layer for the TSCH protocol, and for AB-TSCH with denylists (DL) of sizes 7, 9, 11, 13 and 15, both protocols in star topology. The PDR in the application layer considers all aspects, including packet retransmissions and transmission failures of EBs. In Figure 24 (b), the PDR is shown in the MAC layer, which considers only the packets that were effectively transmitted, disregarding the dropped packets due to problems in reception of EBs or synchronization failures. This allows analyzing the quality of the links in only one direction, from the end nodes to the sink. In Figure 24 (c), the values relative to the RNP are shown, which is the average number of attempts made for each data packet generated in the network,

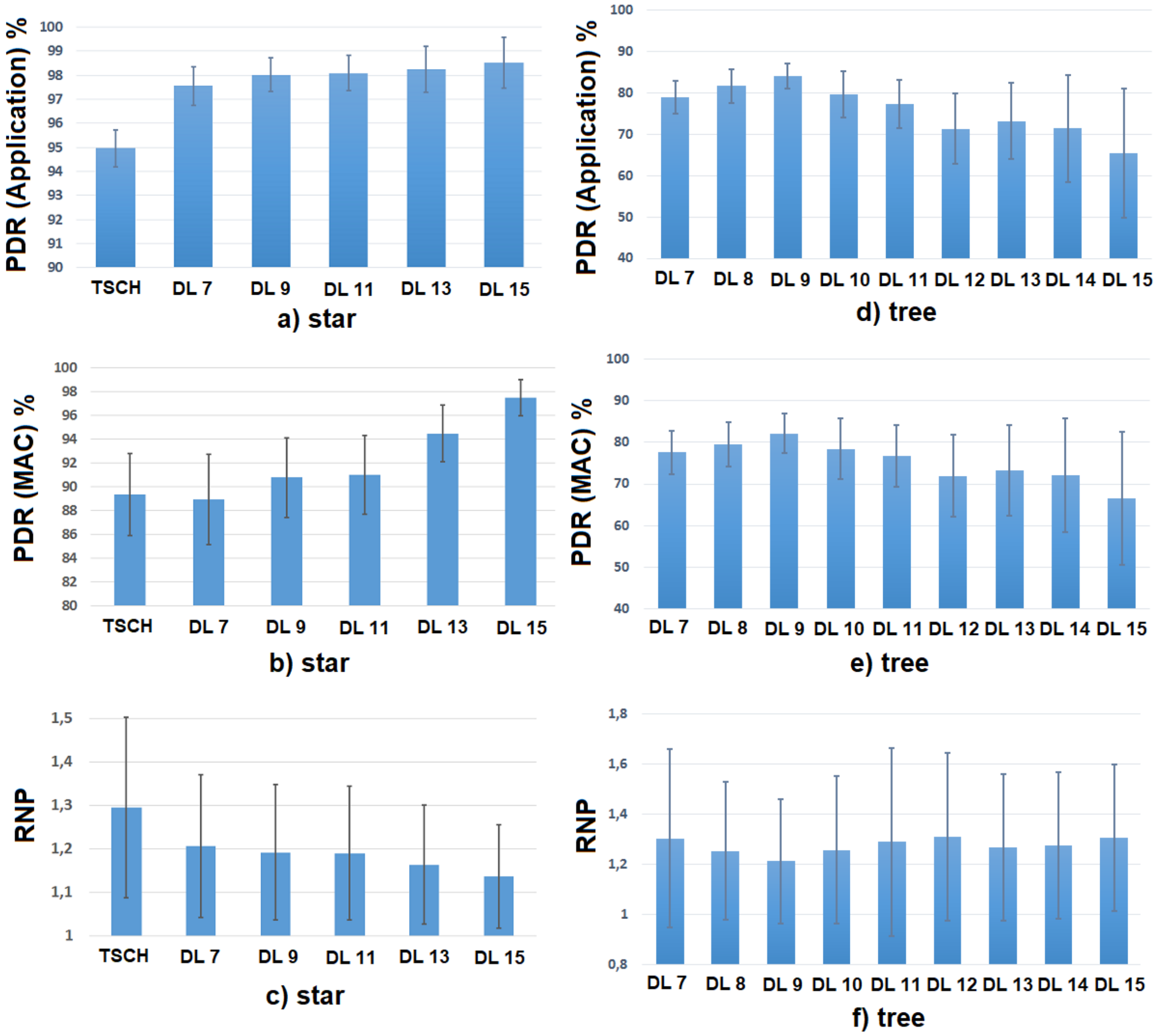

Figure 24: Results from the star topology: a) *PDR in Application layer*, b) *PDR in MAC layer* and c) *RNP*. Results in tree topology: d) *PDR in Application layer*, e) *PDR in MAC layer* and f)*RNP*.

regardless of reception. An attempt of transmission and one of retransmission were used, in case of not receiving the packet confirmation by the sink node, totaling two as the maximum value of the RNP.

Channel adaptation is a frequency hopping method in which the device remains on the same channel for a long period as long as the channel is of good quality, and the hopping is performed only when the channel quality reduces enough to the point of losing packets. Therefore, the number of hops across the available spectrum in this method is smaller than the one proposed by the TSCH protocol, in which at each transmission the device hops to a different channel.

When channel adaptation is used is star topology, which is a slow frequency hopping method, as in the 15-channel denylist scenario, the results show a higher PDR than in the other configurations because the latter use fast frequency hopping. The TSCH protocol in this experiment does not consider the use of LQEs, therefore, it does not use denylists. The other configurations use this estimator, thus the result shows better PDR, since the poor quality channels are blocked.

Concerning the experiments with tree network, depicted in Figure 24 (d), the application layer with denylists between denylist 7 and 10 presented almost the same results, as well as in the MAC layer, depicted in Figure 24 (e). The relevant matter of this experiment is that, the larger the denylist size for tree networks is, the smaller the number of packets received by the sink node is. This can be explained due to the interference of the same channels and adjacent channels caused by some nodes in simultaneous transmissions. With the limited channel diversity, many packets are lost or retransmitted.

Different from the experiment in the star network, the tree network presented almost the same results regarding the RNP for all the scenarios. This can be explained because this experiment was performed using only the AB-TSCH protocol, and there are no noticeable changes in each denylist setting for the protocol. One factor to note in RNP is the high standard deviation in all denylist scenarios, and because it behaved similarly to the TSCH in star network, with almost the same average number of retries made for each packet generated.

In Figure 25 (a), it is possible to see the empirical cumulative distribution function of the delay in star network, and in (b) for tree network. The delay is regarding the time interval between the reception of two consecutive packets for both protocols. Packets are generated at a rate of 1 packet/s at the application layer, but the delay to gain access to the channel and the failures in the transmissions cause variations in the time intervals

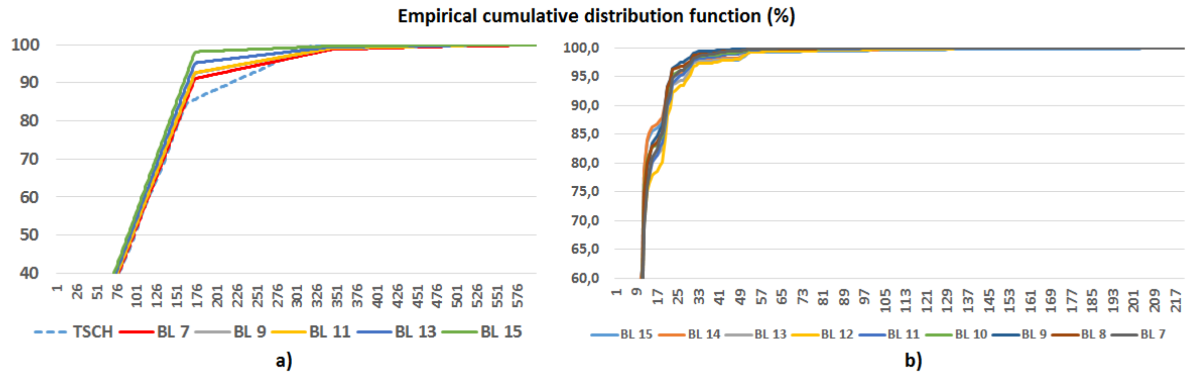

Figure 25: Empirical cumulative distribution function of the delay in ms in star (a) and tree (b) topologies

between two packet receipts.

The graphic with the delay information only considers the packets actually received. The access delay is lower for the AB-TSCH protocol, since approximately 97% of the packets were delivered with a delay of less than 172 ms. In the TSCH, around 90% of the packets were delivered with a delay of less than 224 ms. In the star network with 16 end nodes, the slotframe in the TSCH lasts 160 ms, and in AB-TSCH, which considers the time interval used to transmit the beacon, lasts 170 ms. In AB-TSCH, 99.5% of the packets were delivered with a delay of less than 340 ms, which is equivalent to two slotframe periods.

When considering the empirical cumulative distribution function of the delay in tree network in Figure 25 (b), since this experiment was performed using the same protocol, but with different sizes of denylist, the results showed almost the same values for all the sizes of denylists. If compared to the star topology, the tree topology showed better performance when, for example, 98% of the packets were delivered with a delay of less than 40 ms.

Other experiments were carried out, and in this time five replications were performed. The replications consider the problem of simultaneous transmissions in tree topology, i.e., the results of AB-TSCH are collision-free, and were compared with the standard TSCH, which does not use LQEs.

Each replication lasted five hours with all the nodes in static positions, and in each experiment the nodes randomly define the place to stay, to ensure that the communications topology changes and the results are different as well. At the end, an average of the performance is calculated for the five experiments, and the result is illustrated in Figure 26.

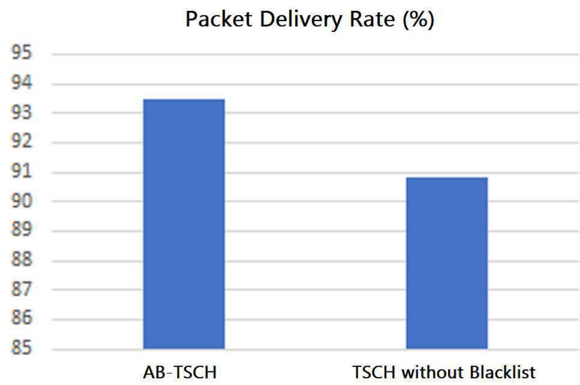

Figure 26: Average PDR of AB-TSCH with channel denylist and no collision, and the TSCH with no denylist and no collision.

Figure 26 shows the PDR of the AB-TSCH protocol and the TSCH with the channel offset management proposed by AB-TSCH to avoid collisions, but without the denylist approach. AB-TSCH performed around 3% better than TSCH, and this shows the importance of using channel denylists, even decreasing network capacity.

The issue of how collisions are avoided in simultaneous transmissions is performed by assigning different channel offsets to each link. In the case of the communication between CH and sink node, only one offset was assigned, because although they use the denylist approach, there is no simultaneous transmission. The challenge arises in communications between end nodes and CHs in which it is possible for one cluster head to receive packets at the same time as the other cluster head.

The use of CHs helps expand the network while allowing greater control of variations in the channel quality from the end nodes, besides allowing to separate the network by collision domains. The LQE function performed by the sink node and CHs frees end nodes from performing this task, leaving them only the task of environment sensing and data transmission.

Assigning multiple offsets throughout the network to choose an allowlisted channel seems to cause higher overhead, so in AB-TSCH this calculation is limited to a maximum of eight attempts. The larger the denylist, the less likely that an allowlist channel to be chosen, the more the offset calculation, the higher the power consumption.

Figure 27 shows the results of a comparison between the AB-TSCH protocol with and without collision avoidance. Figure 27 a) shows the PDR in each experiment, and Figure 27 b) shows an average of both approaches. It is important to note the importance of a proper channel offset management. The experiments without this management performed worse,

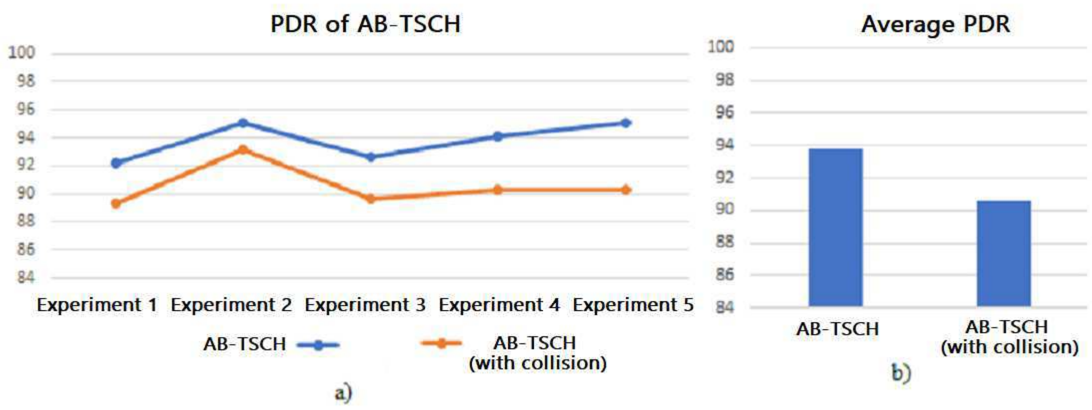

Figure 27: a) PDR of AB-TSCH in two approaches, the first with collision avoidance, and the other subject to collisions; b) Average PDR from the five experiments.

very similar to the TSCH without denylist in Figure 26.

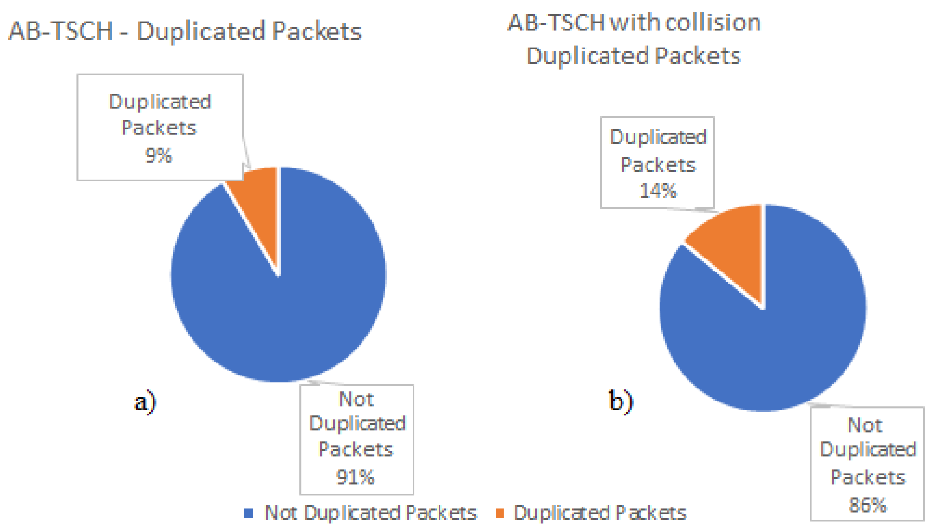

Figure 28: a) Duplicated packets of AB-TSCH; b) Duplicated packets of AB-TSCH without the channel offset management.

Figure 28 shows the result of duplicated packets for each approach. The duplicated packets are accounted for to estimate the link quality in both sides (end node to root and vice versa) for a better estimation. In AB-TSCH, 9% of the transmitted packets were duplicated while in AB-TSCH with collisions 16% of transmitted packets were duplicated. If proper channel offset management is not done, the likelihood that nodes will receive duplicate packets increases, which adversely affects performance and power consumption.

4.3 Final Considerations

This chapter introduced a new MAC layer multichannel protocol called AB-TSCH, which uses the frequency hopping method and the function of LQE for monitoring the link quality of the 16 channels defined by the IEEE 802.15.4 standard. The proposed protocol assigns dynamically channels to the devices in each transmission, based on the value obtained from the real-time estimation of the quality of the links.

The denylist resulting from the evaluation performed by the estimator is sent to the devices through data and ACK packets. By not using EBs for this purpose, this approach prevents the denylist to be out of date when reaches the device, in case the EB is not received on time.

The AB-TSCH protocol was compared with two other approaches, one without considering the channel offset management, which leads to collisions, and the TSCH mode without the denylist approach. By default, there is no specification in the TSCH on how the denylist should be conducted, although its use is an alternative. The protocol outperformed these approaches in terms of PDR, and determinism.

The average PDR in the application layer remained close to 95% for all the five scenarios, even considering the variations in the quality of the channel over time. The approach of using all the 16 available channels for the EBs is a good solution to deal with spatial variations in the quality of channels, instead of using a single control channel.

Chapter 5 presents another approach based on AB-TSCH, and uses a triple list of channels instead of two, and do not flush the channel denylist. As discussed in related works, none of them used the idea of a triple channel list. It was evaluated using the same structure and scenarios of the one described in this chapter, including a larger network with four cluster heads.

5 Triple List of Channels Using Fuzzy Logic

In this chapter, the use of three lists of channels to manage channel assignment is proposed: a denylist to indicate channels that should not be used, a allowlist of channels with good quality, and a greylist of channels with uncertain quality. A fuzzy logic method to classify the channels and include them in the appropriate list is proposed, as well as a suitable channel offset management mechanism to avoid collisions.

5.1 Fuzzy Logic for Channel Lists Management

As mentioned, when referring to the list of channels, the most used concept is that of blacklist and, as opposed to blacklists, there are whitelists. However, the concept of blacklist or denylist has a shortcoming; in a whitelist, or allowlist, there is no doubt that a channel is good, but in a denylist there may be some doubt whether that channel is really bad because, if a defective quality estimator is used, it is possible that a channel with intermediate quality be denylisted.

Many concepts in the real world cannot be well represented using clearly defined boundaries. L. A. Zadeh developed the fuzzy set theory, which generalizes the classic set theory to allow objects to have degrees of membership to certain sets, allowing the representation of vague and imprecise concepts, while maintaining mathematical precision in the treatment [Zadeh 1965].

Most problems encountered in real life can be solved based on inaccurate, incomplete, and vague information that is available at the moment, and yet people are able to solve these problems satisfactorily. This is due to the approximate reasoning process from this information, to obtain an approximate result, but satisfactory for that problem.

Formally, a fuzzy set A of the speech universe Ω is defined by a membership function

$\mu_A : \Omega \rightarrow [0, 1]$. This function associates with each element x from Ω the μ_A(x) degree to which $x \in A$[1]. The membership function μ_A(x) indicates the degree of compatibility between x and the concept expressed by A:

- μ_A(x) = 1 means x is completely compatible with A;
- μ_A(x) = 0 means x is completely incompatible with A;
- $0 < \mu_A(x) < 1$ means x is partially compatible with A with μ_A(x) degree.

A fuzzy control system links fuzzy variables using this set of rules. A fuzzy variable has a crisp value which takes on some number over a pre-defined domain (universe), and the rules are simply mappings that describe how one or more fuzzy variables relates to another.

Expressed in terms of an *if-then* statement, in which the 'if' part is the antecedent and the 'then' part is the consequent, the problem tries to assess whether a channel should be used or not, depending on the metrics provided to estimate the link quality.

In the domain of WSN, there is a high inaccuracy to define when a network is of good or poor quality. More precisely, there may be trends represented by curves, and one of these curves can be defined using a Gaussian distribution, for example.

The Gaussian distribution is a continuous function which approximates the exact binomial distribution of events. If the number of events is very large, then the Gaussian distribution function may be used to describe physical events. It is assumed that during any measurement values will follow a normal distribution with an equal number of measurements above and below the mean value.

In the mechanism proposed in this chapter, the protocol analyzes the channels and assigns each channel to one of the three available lists. The definition of lists uses a fuzzy set approach, and the channels with low quality are denylisted and set as temporarily unavailable to be used. The channels with uncertain quality, which are not so bad to be denylisted, nor so good to be allowlisted, are then greylisted. The allowlist is dedicated to channels that present good quality.

In the example of this work, as illustrated in Figure 29, three metrics are used to define the channel list where a given channel wil be temporarily inserted into. The proposal's control system establishes that the higher the PDR, the higher the percentage of success and the channel has high quality. When the PDR reaches around 80%, there is a doubt about the channel quality if it is high or acceptable.

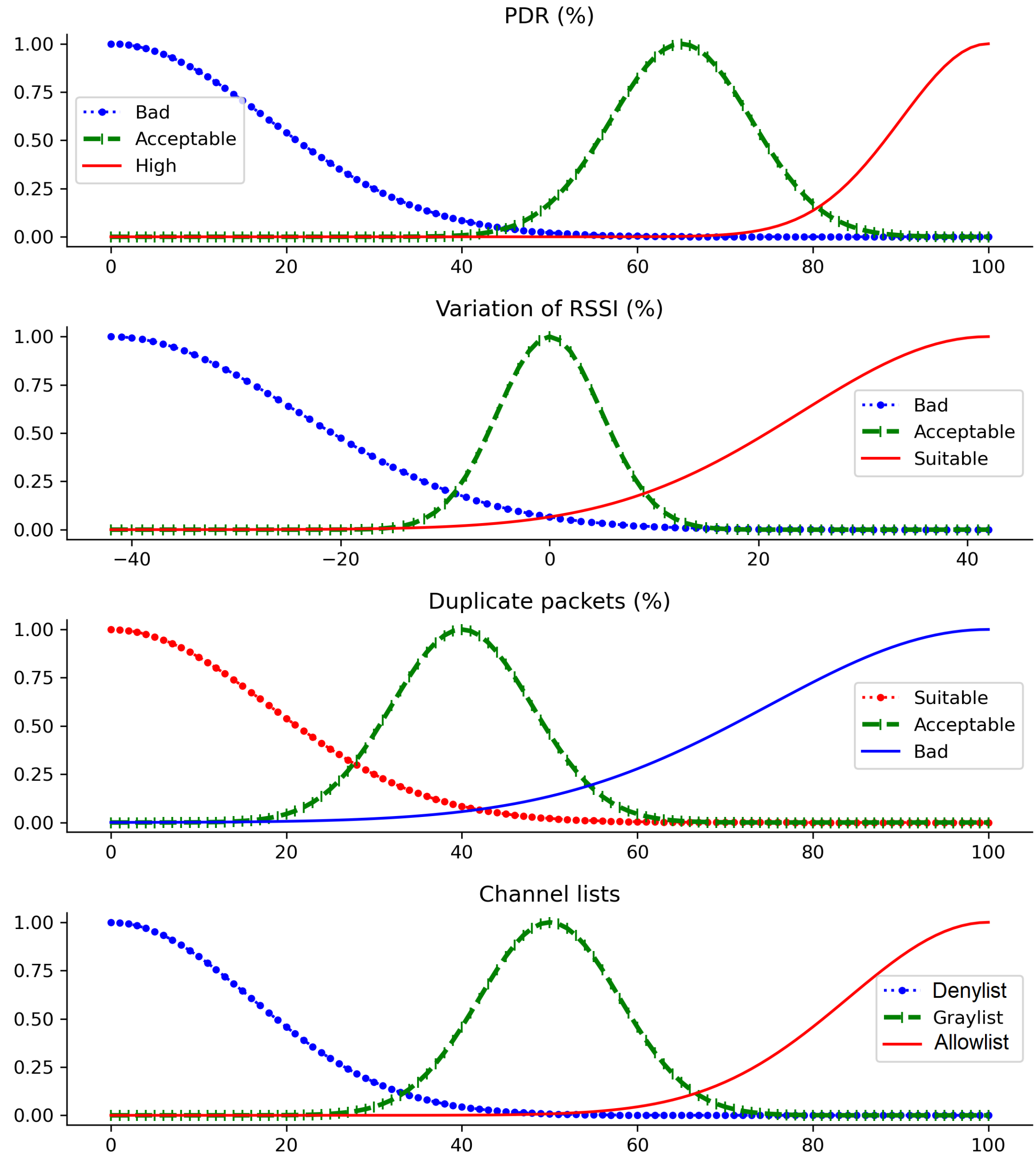

Figure 29: Fuzzy control system in WSN with PDR, the variation of the RSSI, and the number of duplicate packets as the input, and the channel lists as output.

The same occurs in the other metrics. The variation of the average RSSI value in relation to the best average RSSI value in the previous cycle is the second metric. The average RSSI value is referring to the moment when the LQE receives 100 packets. If in the next cycle the average value is reduced, it indicates that there was a deterioration in the link quality in relation to the previous period. If this variation is above the value established for the network, according to the fuzzy logic method, it is possible that the channel has changed its profile and is now considered acceptable or even of bad quality.

In very dynamic environments, it is possible to reduce the number of packets to be received by the LQE, so that the estimation is carried out more frequently and it does not take so long for the LQE to realize that a given channel has low quality.

The last metric is related to the number of duplicate packets. When an end node sends a packet to the cluster head, but does not receive an ACK from it to confirm that it was received, the end node understands that the packet probably was not received by the cluster head and re-sends the same packet in the next timeslot. In some cases, the cluster head actually received the packet, sent back the ACK packet to the end node but it did not receive it. When receiving the same data packet in the next timeslot, the cluster head understands that it has already been sent and considers it as duplicated. This may indicate a problem with the downstream direction of the link, and for the network to have high performance, both sides of the communication need to present good quality, not only the upstream direction. In the example of Figure 29, if around 27% of transmitted packets were duplicate, there is a doubt about if the profile of the channel is suitable or acceptable, and around 70% there is no doubt that the channel is presenting bad quality concerning duplicate packets.

The decision of which list the channel will be inserted into is depicted in the last example of Figure 29. The mean and standard deviation (sigma) values that creates the curves of the Gaussian function were based on the characteristics of the network and the environment and were used according to Table 9. To precisely define these values, it would be necessary to study the environment to be monitored beforehand.

If the characteristics of the network or the environment change over time, this mechanism could adapt to the new reality, as it is based on the average quality of the previous period. However, it is also possible to reprogram the sensors so that the fuzzy logic parameters are updated in the event of a substantial change in network characteristics, such as the change of network monitoring location.

The channel is considered to have good quality if it presents not only a positive variation of RSSI in relation to the last cycle, but it is also based on the PDR and the number of duplicate packets metrics together. If the quality level was good in the previous cycle, maintaining the same level or even reducing it a little could still classify it as a good channel, as it also depends on the other two metrics.

To model this problem, there is an entry for the system, also called antecedent, which is formed from the PDR, the variation of the RSSI, and the number of duplicate packets, the consequents, i.e., the output with the channel lists, and the decision rules are defined

Metric	Profile	Mean	Sigma
PDR	Bad	0	18
	Acceptable	65	8
	High	100	10
RSSI Variation	Bad	-42	18
	Acceptable	0	5
	Suitable	42	18
Duplicate Packets	Suitable	0	18
	Acceptable	40	8
	Bad	100	25
Channel List	Denylist	0	16
	Greylist	50	8
	Alowlist	100	16

Table 9: Mean and standard deviation to generate the Gaussian fuzzy membership function for the PDR, the variation of the RSSI, and the number of duplicate packets.

as follows:

- **Antecedent (input)**
  1. **PDR**
     - Universe: Scale of 0% to 100%;
     - Fuzzy set: bad, acceptable, high;
  2. **RSSI variation**
     - Universe: Scale of -42% to 42%;
     - Fuzzy set: bad, acceptable, suitable;
  3. **Duplicate packets**
     - Universe: Scale of 0% to 100%;
     - Fuzzy set: suitable, acceptable, bad;
- **Consequent (output)**
  - **Channel list**
    * Universe: Scale of 0% to 100%;
    * Fuzzy set: denylist, greylist, allowlist;

The universe set consists of PDR, RSSI variation and the number of duplicate packets. On a scale of 0 to 100%, the PDR reached in the channel by a given device is defined. The scale of RSSI variation is dependent on the intensity and frequency of RSSI variations, and for the studied environment, a positive variation ranges from 0 to 42%, and a negative variation from -1 to -42%. In a network with fewer variations, this range can be reduced. Regarding the number of duplicate packets, a value between 0 and 100% of the sent packets to be duplicated; the higher this percentage, the lower the quality of the network.

The universe of the consequent part is related to the overall quality of the network and its respective channel lists. The closer to 100%, according to the decision rules, the higher the probability that a given channel will be inserted into the allowlist, and the lower this percentage, the higher the probability of being inserted into the denylist. The question is whether the channel is neither good nor bad enough, so it is likely to be inserted into the greylist.

- **Decision Rules**
  - IF the *PDR* was high and *RSSI variation* was (suitable or acceptable) THEN the *Channel list* will be allowlist
  - IF the *PDR* was bad or *RSSI variation* was bad THEN the *Channel list* will be denylist
  - IF the *PDR* was bad and *RSSI variation* was acceptable THEN the *Channel list* will be denylist
  - IF the *PDR* was (high or acceptable) and *RSSI variation* was (bad or acceptable) THEN the *Channel list* will be greylist
  - IF the *PDR* was acceptable and *RSSI variation* was (acceptable or great) THEN the *Channel list* will be allowlist
  - IF the *PDR* was (acceptable or high or bad) and *Duplicate packets* was bad THEN the *Channel list* will be denylist
  - IF the *Duplicate packets* was acceptable and *RSSI variation* was (acceptable or suitable) THEN the *Channel list* will be greylist
  - IF the *Duplicate packets* was suitable and *RSSI variation* was bad THEN the *Channel list* will be allowlist

In fuzzy logic, mapped using crisp values, both variables are formed from adjectives such as 'Bad', 'Acceptable', 'High', 'Suitable'. In general, fuzzy concepts are widely used by humans to communicate the impression of things, whether something is good or bad. However, given the need to create a computational system that does not work with fuzzy logic, it is necessary to transform these concepts into numbers.

INPUTS			OUTPUTS	
PDR	RSSI Variation	Duplicate	Universe	Fuzzy set
85%	-10%	0%	45.8171	Greylist
80%	-5%	20%	53.6162	Greylist
55%	-18%	80%	32.1620	Denylist
90%	0%	70%	51.9774	Greylist
95%	3%	15%	66.7860	Alowlist

Table 10: Example of simulated values of PDR, RSSI variation and duplicate packets to obtain the result on which list the channel will be inserted into

The fuzzy logic ends up losing important information that is embedded in these adjectives, but the computer tries to take advantage of it somehow. A membership function is then created that will attempt to capture the meaning of that adjective. The fuzzy controller is modeled using these adjectives, allowing to simplify the system design.

To simulate the usage of this system, Table 10 shows some inputs and outputs, using crisp values and the fuzzy set. The simulation was performed with a python fuzzy logic toolbox called *scikit-fuzzy*.

5.2 Description of the Proposal

Besides star topology, and tree topology with two cluster heads studied in Chapter 4, a larger topology was studied with four cluster heads and 48 end nodes, in which each cluster head is directly connected to 12 different end nodes, as depicted in Figure 30. This means that in the topology with two cluster heads, two links (4 end nodes) have parallel transmissions in each timeslot, and in the topology with four cluster heads, four links (8 end nodes) have parallel transmissions.

Tree topology allows to use the concept of cluster heads that divide the network into subnets, in which each cluster coordinates, work as routers and as LQE for the links in the specific subnets. In addition to these advantages, the cluster allows lower energy consumption of end nodes, since its use means that the end nodes have the single task of monitoring the environment and sending the data to the destination.

Star topology, studied in a previous work with channel denylists [Queiroz et al. 2018], can also be used with this proposal. In that work, the larger the channel denylist, the higher the network performance as only the channel with the best quality of each link would remain. Since there were no simultaneous transmissions, and in each timeslot there was only one data transmission directly to the sink node, the solution presented high PDR and determinism.

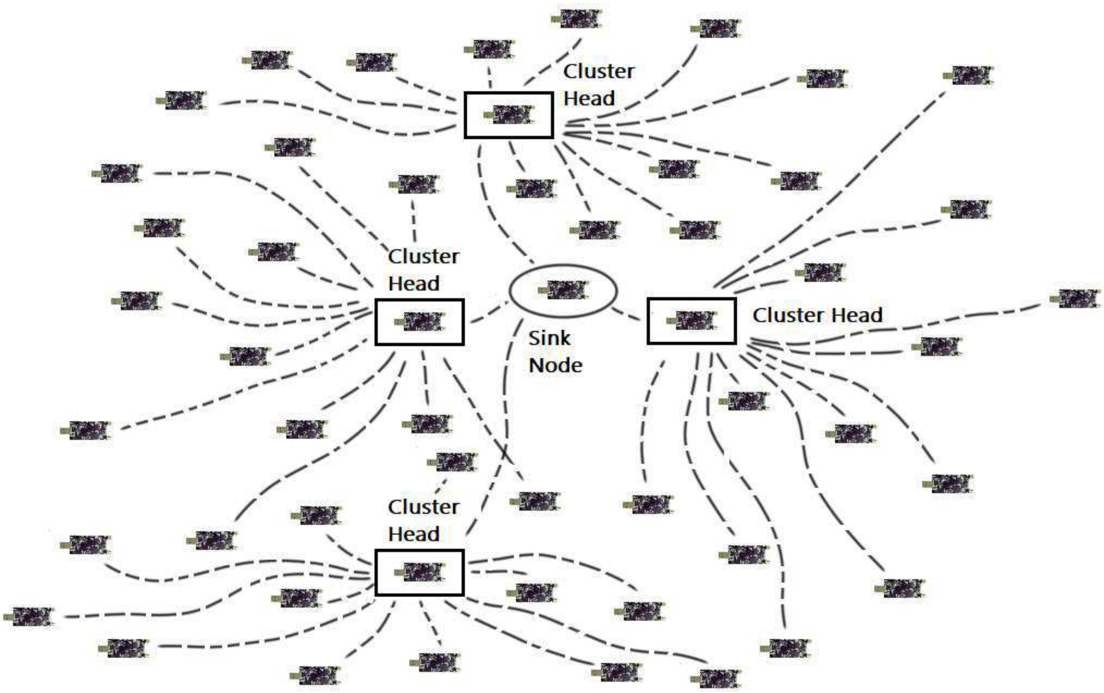

Figure 30: One of the topologies with four cluster heads.

In this proposal, the medium access is based on a structure called multislotframe, similar to the work in [Gomes et al. 2019], which proposes this multi-slotframe structure and was adapted to be compatible with TSCH, and similar to the AB-TSCH approach.

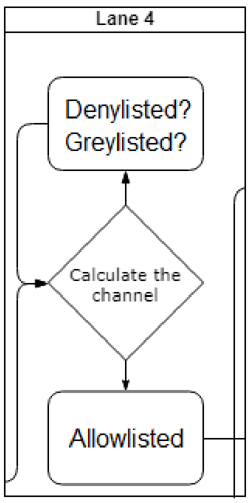

Figure 31: Lane 4 of the diagram for triple channel list.

The diagram of the end nodes steps is very similar to the AB-TSCH, and includes in Lane 4 the analysis of the greylist, as depicted in Figure 31. The slotframe structure with EBs in tree topology is the same as the AB-TSCH protocol.

An analysis was carried out to define the number of timeslots allocated to the cluster heads to forward the data packets in each slotframe, in order to achieve a good compromise relationship between the number of packets that a coordinator can forward to the sink node, per second, and the duration of the slotframe, which influences the network delay. The resulting timeslot structure is depicted in Figure 32 in a network with four cluster heads.

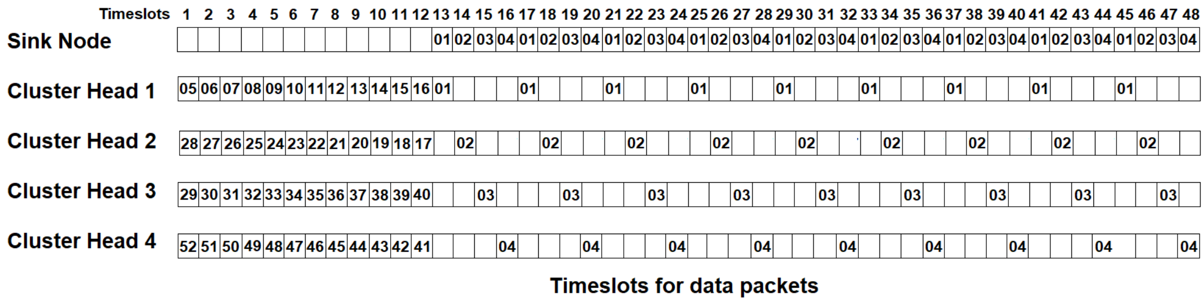

Figure 32: Timeslots for data transmission in topology with four cluster heads.

In this figure, 48 timeslots are assigned for data packets, and the first 12 timeslots are assigned for the cluster heads to receive data packets from the end nodes. For example, in the first timeslot, the end node identified by number 05 transmits a packet to cluster head 1, the end node 28 transmits a packet to cluster head 2, and so on. The white squares indicate that the transceiver is off, therefore in the meantime, the sink node keeps its transceiver off to save energy up to timeslot 13, in which it receives the data packet forwarded by cluster head 1. Timeslot 14 is dedicated to receiving the forwarded packet from cluster head 2 and so on, up to timeslot 17, in which this process is restarted until timeslot 48.

On average, the cluster head can forward X packets per second from each end node directly connected to it. If X is less than the packet transmission rate of the end node multiplied by the maximum number of transmission attempts per packet, some packets can be dropped at the coordinator. On the other hand, the higher the value of X, the larger the delay in the network.

In the example, 48 end nodes transmit data packets, and there are four cluster heads to forward these packets. Normally 12×4 timeslots would be necessary for the cluster heads to forward the data to the sink node, in addition to the 12 timeslots dedicated to the end nodes to transmit these packets to the cluster heads, totaling 60 timeslots. However, as a way to decrease the delay with an acceptable packet loss, the following strategy was used, which resulted in only 48 timeslots allocated to the network. This is for a network with simultaneous transmissions on different channels, and if there were no simultaneous transmissions, the delay would be even greater with 12×4 timeslots for end nodes, and 12×4 timeslots for cluster heads, totaling 96 timeslots.

As in the experiments there are 48 end nodes, and each one has 10 ms to send data packet and receive an ACK packet, the slotframe size for the data is 480 ms. To calculate the value of X, the Equation 5.1 based in [Gomes et al. 2017] was defined to know the number of timeslots allocated to coordinators:

$$N_{ts} = (SF - (2 \times T_{ts}) - (D_{end} \times T_{ts}))/(T_{ts} \times N_{ch}), \tag{5.1}$$

in which N_{ts} is the number of timeslots allocated to coordinators, SF is the slotframe duration in milliseconds, T_{ts} is the timeslot duration in milliseconds, D_{end} is the number of end nodes connected to each cluster head, and N_{ch} is the number of cluster heads. In this the example, the result is shown in Equation 5.2:

$$N_{ts} = (480 - (2 \times 10) - (12 \times 10))/(10 \times 4), \tag{5.2}$$

in which approximately 9 timeslots should be allocated to the coordinators. This value is used to calculate the X in the Equation 5.3:

$$X = N_{ts}/(D_{end} \times t \times (SF \times 10^{-3})), \tag{5.3}$$

and the result shows that 1.5625 packets can be forwarded by each cluster head, as exemplified in the Equation 5.4.

$$X = 9/(12 \times 1 \times (480 \times 10^{-3})). \tag{5.4}$$

Thus, the values of X equal to 1.56 packets per second per final node are sufficient to maintain a good quality of service in this case.

Concerning the fiels of packets, every data and ACK packet transmitted has a field of denylisted and greylisted channels that the transmitter and receiver are not allowed to use (denylist) or may use only on later transmissions (greylist).

Figure 33 illustrates the structure of the data packet transmitted by the end nodes, and it is very similar to the structure of AB-TSCH, and includes the channel greylist field with 2 bytes. The channel lists are inserted into the ACK packets transmitted by the receivers (LQEs) and into the data packets transmitted by the nodes. Initially, the data packet has only allowlisted channels, keeping the bits zeroed in the deny and greylist fields, until the analysis is done by the LQE and the lists are informed in the ACK packet and established in the next data packet.

The greylisted channels are used only if the network in general presents abruptly

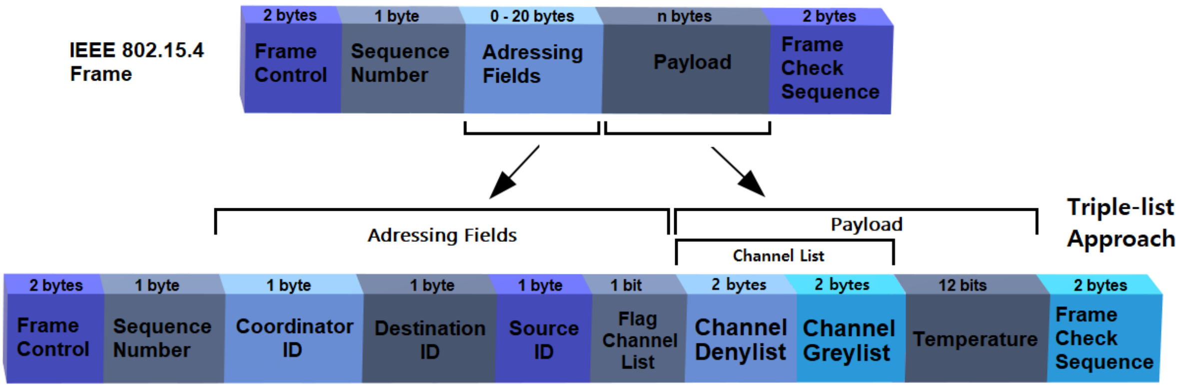

Figure 33: Data packet from IEEE 802.15.4 standard and the proposal of this work.

reduced quality, or if the number of allowlisted channels is insufficient to provide the principle of transmission on different channels. This number depends on the number of devices with simultaneous transmissions. For example, if there are 16 channels available in the standard and 4 of them are denylisted, there are 12 channels left for the devices to use in the allowlist or greylist. In the worst case in which there is an abrupt drop in the quality of the channels, and those used in the allowlist are no longer of good quality, the channels in the greylist will be used, and if there are no channels in the greylist, the denylisted channels will be used. This is the alternative to circumvent the negative effect on several channels in the allowlist at the same time.

If more than one channel are of poor quality, in addition to the four that were denylisted in the previous period, the oldest ones are replaced and then greylisted, and the latest poor channels are denylisted. Throughout the network, certain channels will present good quality to the point of being allowlisted, and may be greylisted if the quality is reduced. Then, they are considered to show sufficient quality to do not be denylisted. The denylist has up to four channels, and as soon as this number is reached, there will always be four denylisted channels, except if there is a sudden drop in the quality of the channels, previously described as the worst case. In this case, it is possible that the denylist is temporarily flushed.

After the first analysis of the LQE, some channels are denylisted, others are greylisted, and the receiver inserts into the ACK packet the information of the channel lists after it receives the data packet from the transmitter. Once the latter receives the ACK packet, it updates its own channel list field, and transmits the new data packet with the updated information and the flag 1_2 (Flag Channel List) activated. The objective of this flag is to inform that the node is aware of the updated channel lists, and once the destination receives it, the transmitter deactivates it. The destination then compares

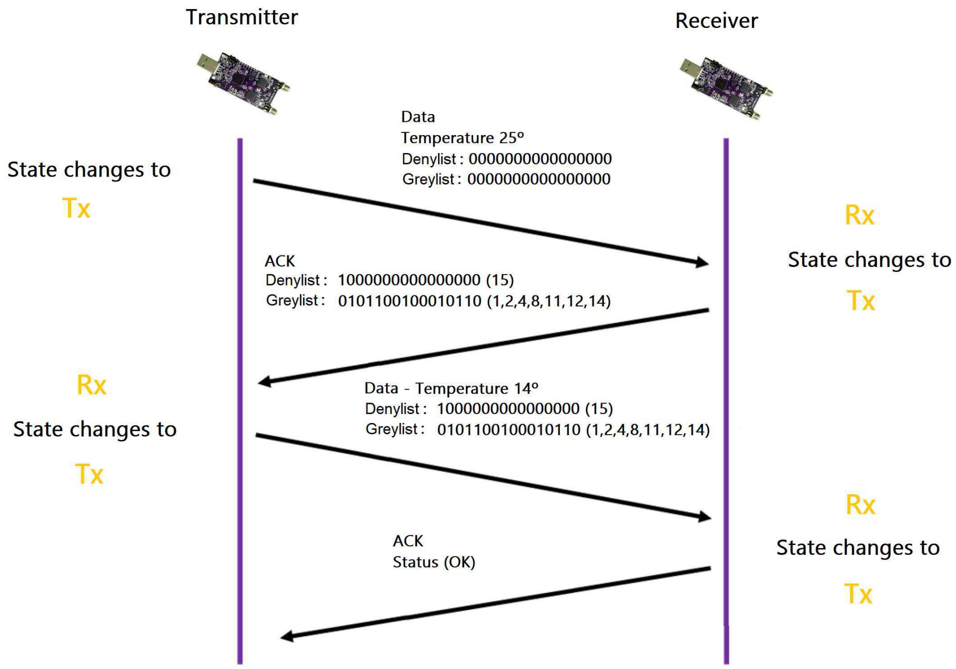

Figure 34: Procedure of exchanging the channel list information between transmitter and receiver.

with the latest ACK packet to know if the transmitter is aware of the channel list. If it matches, the receiver then sends the ACK packet to inform the transmitter that now they can communicate using the new channel list. As noted, the procedure is very similar to the procedure of AB-TSCH.

Figure 34 shows an example of it, in which at first, all channels are allowlisted, and after the LQE analysis, channel 15 should be denylisted, channels 1, 2, 4, 8, 11, 12, 14 should be greylisted, and the rest remains allowlisted.

In the solution proposed in this thesis, if four channels are already denylisted, and there are more channels with low quality, the oldest ones will be removed and greylisted, and will be replaced by the new channels with low quality, so that the number of those denylisted does not exceed four.

Algorithm 1 shows the channel assignment if the proposal, in which line 1 assigns to the variable *channel* the frequency resulting from the Equation 3.1. Then the counter indicating the position of the channel offsets array is initialized. The *while* and *if* structures are used with bitwise operators, in which they analyze whether the assigned channel is denylisted or greylisted. If the channel is not allowlisted, another channel will be chosen by increasing the *contOffset* variable, until it finds a allowlisted channel to be used by the link. At this stage, the links are already aware of the three channel lists and are just

Algoritmo 1 Channel Assignment with Channel Offsets

```
channel = FHS[(ASN + CO) mod FHS_L]; // TSCH Equation
contOffset = 0;
enquanto ((denylist[endnode]>>channel)&1) OR ((greylist[endnode]>>channel)&1) faça
    // If the channel is allowlisted, it will be used and do not do anything.
    se ((!(denylist[endnode]>>channel)&1) AND (!(greylist[endnode]>>channel)&1) então
        break;
    senão
        // The resulting channel is not allowlisted. Choose other with the next channel offset
        contOffset++; // Array of channel offsets of the Cluster Head 1
        channel = (ASN + offset1[contOffset]) % 16;
    fim se
fim enquanto
```

looking for a suitable channel to perform the hopping.

5.3 Simulation Setup

Simulation time	2 hours
Number of channels	16 channels
Number of end nodes	48 end nodes
Number of cluster heads	4 cluster heads
Number of sink nodes	1 sink node

Table 11: Parameters used in the simulations.

Table 11 shows the parameters used for the simulation, and for each configuration, 3 replications were generated and lasted 2 h. In each one, the position of the nodes was assigned randomly, with distances less than 60 m from the coordinators. The experiments evaluated networks with and without channel denylists, with channel triple list, with and without collisions. Concerning the configuration with collisions, all the nodes with parallel transmissions send packets on the same channel (they use the same channel offset). In the 4-cluster head topology, four end nodes transmit at the same time on the same channel. Concerning the configuration without collisions, all the nodes transmit on channels spectrally distant from each other.

To better understand the behavior of the simulation model, Figure 35 shows the reception power in RSSI of one node in the 4-cluster head topology using the triple channel list in two different configurations of the average change time, that lasted 30 minutes, i.e., highly dynamic environment, and that lasted 100 minutes (1 hour and 40 minutes), i.e., less dynamic environment.

It is possible to notice that in the first 25000 samples, the reception power was the same in both configurations, and after that the 30-minute configuration increases the

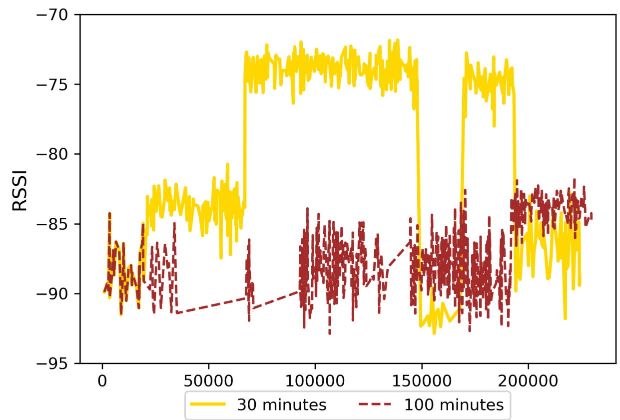

Figure 35: Samples of received power of a node in a given channel with the average change time of 30 and 1 hour and 40 minutes.

quality of the channel while the 100-minute configuration remains almost unchanged, i.e., with poor quality.

Next to the 50,000 samples the 100-minute configuration decreases the channel's quality and it is then denylisted. Only after the 50,000 samples the channel was used and again it is denylisted. It was only reused when close to 100,000 samples with a little better quality. A little before 150,000 samples the channel is denylisted due to several duplicate packets, and its quality is only improved (around -85 dBm) next to 200,000 samples.

The 30-minute configuration presents several modifications in the channel quality. To assign a list to a link, in our solution the LQE has to wait 100 packets, and in the meantime it is possible that certain links present temporary deep fading as in the example of the sample around 150,000, were the channel reduced its quality about 20%, but it was not denylisted. The LQE began to count the next 100 packets when it was presenting very good quality (around -73 dBm), and the average value when it reduced its quality (-92 dBm) is then about -82 dBm. Therefore, the difference considered by the LQE was about 12% worse than it was when presented very good quality, which according to the rules of the fuzzy set, it was not enough for the channel to be denylisted, since the RSSI variation is not the only evaluated metric.

In some cases of the real environment, it is possible for some channels to enter a state of deep fading and remain so for a long time. When this happens, it is possible for LQE to capture this effect and denylist these channels. However, as it is not possible to predict whether those channels will remain in this state for a long time, along the time other channels will present low quality and need to be denylisted. Since the denylist has a maximum of four channels, it is likely that those that are in a state of deep fading will be removed from the denylist (as they would become the oldest on the list), and enter the greylist. As only the allowlist is used by the devices, they would not use those problematic channels, unless the channels of the allowlist also present an abrupt reduction in quality. Since there is a range of 16 channels, the probability that all channels present poor quality is small.

5.4 Results

The nodes were arranged in an area of 200 x 200 m, and three replications were executed for the evaluated configurations. In each replication, the network operated for two hours, and the position of the nodes was randomly assigned. To guarantee a fair result, for each replication the same seed was used (seed is a parameter of the simulator) to evaluate the protocols, and different seeds were used in the different replications. Thus, the protocols were evaluated considering the nodes in the same position and with the same characteristics for the wireless channel during the replications.

Figure 36 shows the performance in terms of PDR in the application and MAC layers of both configurations. The experiments evaluated the TSCH in a blind channel hopping fashion, with channel offsets spectrally distant from each other to prevent sensors with parallel transmissions from sending data on similar or adjacent channels. This is the most used configuration in literature for the TSCH protocol, which does not use channel denylists nor use channel quality estimators.

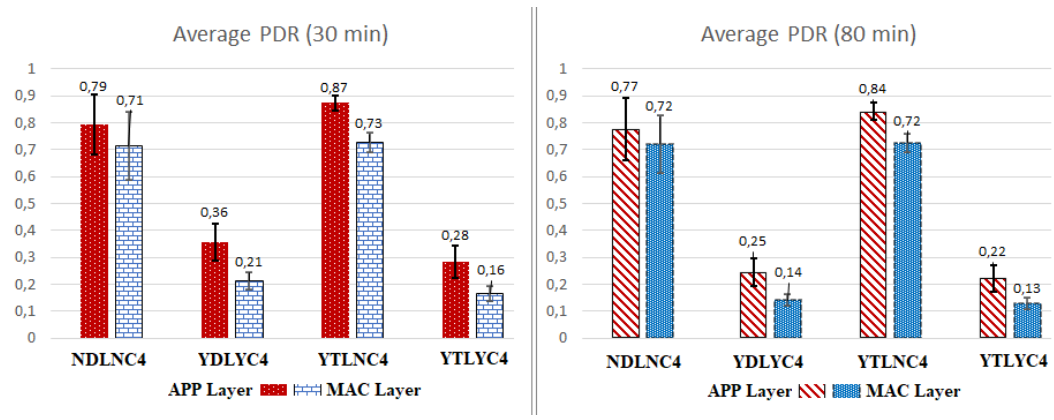

Figure 36: PDR in Application and MAC layers in two different average change times.

It is straightforward that the application layer has higher PDR than the MAC layer. The difference is that the application layer considers packet retransmissions, so the values

Label	Meaning
NDLNC4	Experiment without channel denylists and without collisions with 4 cluster heads
YDLYC4	Experiment with channel denylists and collisions with 4 cluster heads
YTLNC4	Experiment with triple channel list and without collisions with 4 cluster heads
YTLYC4	Experiment with triple channel list and collisions with 4 cluster heads

Table 12: Explains the labels used in the experiments

will be higher. In the simulations executed for this paper, the nodes can perform only one retransmission, if the first attempt fails.

In addition, three other approaches are evaluated, the first one using a channel denylist (double list) with collisions, and the others that represent the triple list approach proposed in this paper, the first one without collisions, and the last one with collisions.

Table 12 shows the labels used in the experiments to identify the simulated configurations, and in each configuration the average change time was set to 30 and 80 minutes. The label NDLNC4 stands for *No Denylist No Collision*, YDLYC4 stands for *Yes Denylist Yes Collision*, YTLNC4 stands for *Yes Triple List No Collision*, and YTLYC4 stands for *Yes Triple List Yes Collision*.

In the study carried out in this work, as mentioned, collisions are avoided by assigning different channel offsets, like what is defined for the WirelessHART standard. This assignment is performed by the network administrator and inserted into the coordinating nodes. However, MAC layer protocols that do not use the concept of channel offset in the frequency hopping need to use other dynamic methods to avoid collisions. In this work, we are evaluating the negative effects they cause when this problem is not considered, although a simple assignment of spectrally different channel offsets highly reduces the negative effects.

The difference in performance is noticeable when the problem of collisions is taken into account, especially in large networks with simultaneous transmissions. The standard deviation is also high for networks that do not use channel lists, either by double list or triple list. In the three approaches that used channel lists, the variation was small compared to the blind approach.

In the experiments without the channel offset management, the PDR in the application layer was very low, from 22% (80 minutes) to 36% (30 mintes). Without collisions, the PDR was between 77% (80 minutes) and 87% (30 minutes), as depicted in Figure 36.

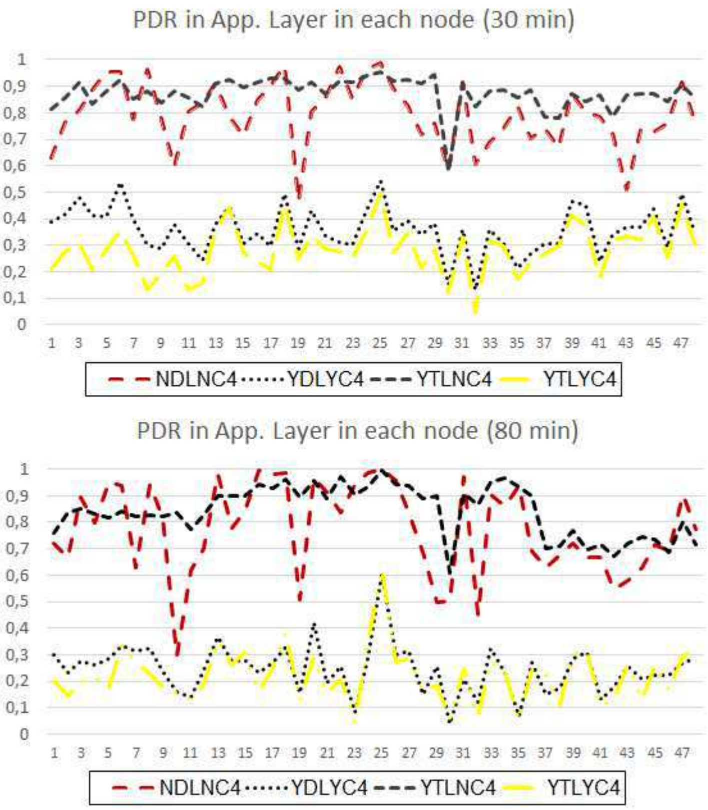

Figure 37: PDR in the Application layer of each node in different average change times.

Figure 37 shows the PDR in the Application layer of each node in different average change times. Although there is no continuity in relation to the graph, since they are data from different nodes, this continuity allows to perceive the difference in the average PDR of each configuration. It is possible to see that a greater variation occurs in networks without link quality estimators (NDLNC4 experiment), because the node hops to channels without knowing their quality, and in a harsh environment such as the industrial one, the probability of hopping to channels with medium or low quality is higher. Using link quality estimators, this probability decreases, and the performance becomes more stable, with a lower variation, such as in the YTLNC4.

Other metric evaluated is the Required Number of Packets transmissions (RNP), depicted in Figure 38. It shows the average number of attempts made for each data packet generated in the network, regardless of reception. For example, if on average each packet is transmitted twice until it reaches its destination, the value of this metric would be 2.

It is possible to notice that the configurations without collision required less attempts of packet transmissions, and the blind approach performed very similar to the triple list approach in both average change times.

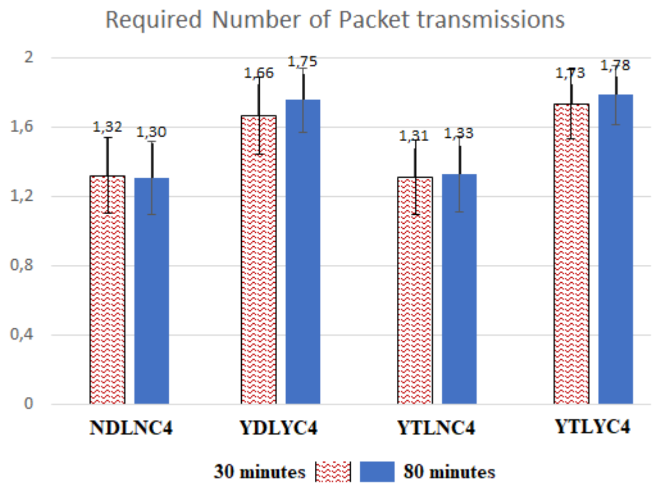

Figure 38: Required Number of Packet transmissions (RNP).

In Figure 39, the empirical cumulative distribution function of the time interval is depicted. The configuration with 30 and 80 minutes performed very similar, therefore only the 30-minute configuration is presented.

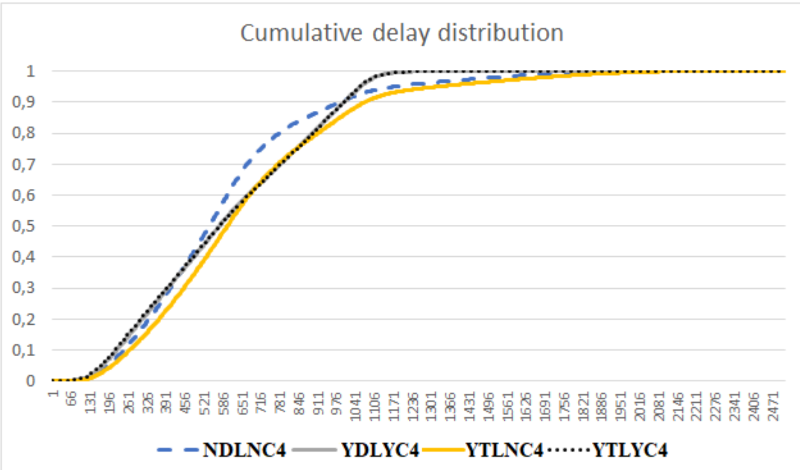

Figure 39: Empirical cumulative distribution function of the time interval in milliseconds.

The time interval is regarding the time between the reception of two consecutive packets. Packets are generated at a rate of 1 packet/s at the application layer, but the delay to gain access to the channel and the failures in the transmissions cause variations in the time intervals between two packet receipts. The graphics with the time interval information only consider the packets actually received.

In YBYC4 configuration, around 95% of the packets were delivered with a time interval of 1056 ms, 1053 ms in YTLYC4, 1174 ms in NDLNC4, and 1320 ms in YTLNC4. In the simulations, the superframe lasts 480 ms, and around 39% of the packets were delivered with a time interval less than this value in YDLYC4 and YTLYC4 configurations, i.e.

with collisions. In the NDLNC4 configuration, 41% of the packets were delivered with time interval less than 480 ms, and in YTLNC4, 33% of the packets were delivered with time interval less than 480 ms.

The access delay is much shorter for the approaches that did not take into account the simultaneous transmissions (YDLYC4 and YTLYC4), followed by the approach without link quality estimation (NDLNC4). On the other hand, several packets can be lost when several nodes transmit at the same time in the case of YDLYC4 and YTLYC4, and several packets can be lost due to frequency hopping in channels with poor quality in the case of NDLNC4.

In this study, all devices have the same priorities, and the maximum number of devices used is directly linked to the number of devices with simultaneous transmissions. As there are only 16 channels available, it is theoretically possible to have 16 cluster heads, and the number of end nodes is limited to the acceptable delay of the environment, according to the network designer.

Figure 40 shows a small part of the code, and describes the main tasks performed by the nodes to transmit data packets and receive ACK packets. In the first part of the code, the cluster head identifies the transmission of a packet directed to it (*getDestination*), and analyzes if the packet is a data packet. If confirmed, it registers this packet to later pass it on to the sink node. With the link quality estimation information, the cluster head inserts this data into its ACK packet, represented in part 2 of the figure (ACK_PACKET). The third part shows the code of the end node, which receives an ACK packet, and retrieves the data from the channel list (*getDenylist*). In the next timeslot, represented in part 4, the end node sends a new data packet ($DATA\_PACKET$) o the cluster head, and this process is restarted.

5.5 Final Considerations

This chapter described a novel approach to assign channels based on a real-time link quality estimation in a channel hopping fashion with timeslots and slotframes. Data and ACK packets are used by the nodes to exchange the information of channel lists, while EB packets are sent by the network coordinators to synchronize the network. The EB packets are sent using the 16 available channels, with no channel list, to maintain the diversity and allows the inclusion of other devices that are not aware of the channel lists.

The channel lists are assigned dynamically every 100 packets received by the sink and

```
---------------------------------------------------------------------------------
    // Cluster head receives data packet
    if (macPkt->getDestination() == cluster_id) {
        if(macPkt->getKindTSCH() == DATA_PACKET) { //DATA PACKET
1)      trace() << "Received packet at cluster head: " << macPkt->getSequenceNumber() <<
        " from: " << macPkt->getSource() << " Power: " << rssi << " channel " << channel;
        }
    }
---------------------------------------------------------------------------------
    // It assigns denylist and then transmit this information in ACK packet
    macFrameAck = new PhDTreePacket("PhDTree ACK packet", MAC_LAYER_PACKET);
    macFrameAck->setSource(SELF_MAC_ADDRESS);
    macFrameAck->setDestination(macPkt->getSource());
    macFrameAck->setSequenceNumber(macPkt->getSequenceNumber());
2)  macFrameAck->setDenylist(denylist);
    macFrameAck->setFlagBL(flag);
    macFrameAck->setKindTSCH(ACK_PACKET);
    setTimer(SEND_ACK, 10); //wait 1ms to transmitt ACK
    toNetworkLayer(decapsulatePacket(macPkt));
---------------------------------------------------------------------------------
    // End node receives ACK packet and retrieves the channel list.
    if(macPkt->getKindTSCH() == ACK_PACKET && macPkt->getDestination() == SELF_MAC_ADDRESS)
3)      denylist_AUX[SELF_MAC_ADDRESS] = macPkt->getDenylist()[SELF_MAC_ADDRESS];
        confirm[SELF_MAC_ADDRESS] = macPkt->getFlagBL()[SELF_MAC_ADDRESS];
    }
---------------------------------------------------------------------------------
    // The next data packet has already the denylist information updated
    if (buffer.size() > 0 && slots[slotCont]) {
        PhDTreePacket *macFrame = new PhDTreePacket("TDMAMAC packet", MAC_LAYER_PACKET);
        packet *p = buffer.front();
        cPacket *pkt = p->pkt;
        int destination = p->dest;
        buffer.pop_front();
        encapsulatePacket(macFrame, pkt);
        macFrame->setSource(SELF_MAC_ADDRESS);
4)      macFrame->setDestination(destination);
        macFrame->setDenylist(denylist2);
        macFrame->setFlagBL(flag);
        macFrame->setAckReq(p->ack);
        macFrame->setKindTSCH(DATA_PACKET);
    }
    trace() << "Transmitting packet " << macFrame->getSequenceNumber() << " from "
    << SELF_MAC_ADDRESS << " channel " << indch << " channel list "
    << denylist2[SELF_MAC_ADDRESS] << " to " << destination;
    toRadioLayer(macFrame);
    toRadioLayer(createRadioCommand(SET_STATE, TX));
---------------------------------------------------------------------------------
```

Figure 40: Codes to transmit data packets to the cluster heads and receive ACK packets.

cluster heads in each channel, according to the rules defined in the fuzzy logic approach. In some environments, a window of 100 packets analyzed may be unreactive, however, the main focus of this paper is presenting the possibility of using three channel lists instead of two, which is the most common approach. The proposed solution shows better performance in terms of PDR and determinism when compared to the other approaches evaluated in this paper, without channel list (blind channel hopping), and with dual channel list (denylist and allowlist).

The results also show that, although presenting higher time interval in comparison to the other approaches, the proposed solution presented better results in larger networks, and in harsh and highly dynamic environments with interference, collisions, and multipath fading.

The WSN need to be implemented considering all aspects that affect the wireless channel, such as shadowing problems, attenuation, spatial variations in the quality of the channels and the non-stationary behavior of the wireless channel for a long term. Multi-channel protocols and dynamic channel allocation alleviate these problems, allowing these networks to adapt to the dynamic characteristics of the channels and to mitigate problems due to interference and effects caused by the multipath profile of the environment.

A mechanism to estimate the link quality in real time on these networks is necessary, without generating overhead in the main application. A new architecture for Industrial WSN was proposed, using channel quality estimators and an appropriate channel allocation approach. Both methods were used in conjunction with the fuzzy logic to define which of the three channel lists the channels should be inserted into. Through this solution, no overhead is imposed on the end nodes of the network and there is no increase in traffic, unlike other solutions found in the literature, which use probe packets or include redundancy in the packets.

One topic to be further investigated is a proper channel offset assignment to mitigate interference together with the local channel list approach. In this case, TSCH defines the channel offset variable in Equation 3.1 so that sensors avoid choosing channels compromised by internal (from neighbors) and external (from Wi-Fi, for example) interference. Some works define a single, and others define several offsets, and this work assigns multiple channel offsets.

The development of a model to evaluate the energy consumption of this approach is planned for the near future. In addition, an experimental evaluation of the proposal will be made in real industrial environments, also using the idea of different priorities for some devices in the network. A study of this approach is also planned using a mesh network with redundant routes. The mesh network increases the complexity of the network and is the third topology supported by the TSCH protocol, besides star and tree topologies.

6 Conclusions and Proposals for the Future

This chapter is organized as follows. Section 6.1 presents the summary of this thesis, and Section 6.3 presents future research directions related to this topic.

6.1 Summary of this Thesis

In this thesis, two implementations were proposed for IWSN to mitigate the negative effects of industrial environments on the wireless media. Two main topologies were studied, with the developed methods, and evaluated using simulations, using a realistic channel model that simulates the behavior of an industrial environment.

In the application of those methods, typical problems of the industrial environment were considered, such as shadowing, attenuation, spatial variations in the channel quality and the non-stationary behavior of the wireless channel in the long term. The use of multichannel protocols and dynamic channel allocation by the assignment of channel lists to improve the quality of service of WSN was discussed. The multi-channel approach presents several advantages if compared to the single-channel approach.

In addition, the need to use quality estimators, that analyze the channels before devices are allowed to use them, has been demonstrated to avoid packet losses and waste of energy.

Given the complexity of adding another element to the network, either by hardware with a dedicated device, or by software, with a built-in function in the coordinating nodes, most studies did not used this approach. However, its advantage is that the LQE is inserted only in the coordinating nodes, leaving the end nodes only to monitor the environment and transmit the captured data to the sink node.

In addition to the PDR, information about duplicated packets on the network and about the variation of the RSSI is also used to estimate the quality of the channel, including in the reverse direction of the link, to identify asymmetry problems. By this solution, no overhead is imposed on the end nodes of the network and there is no increase in traffic, unlike other solutions found in the literature, which use beacon packets or include redundancy in the packets.

An approach was also used to decrease the latency of communications and increase the throughput of the network. This approach causes some packet losses, but it decreases the network delay, allowing devices to send more data to the sink, and thus compensating for small losses by using this approach.

The use of channel lists proved to be positive in relation to the blind approach. In both double and triple lists, the performance was superior to the evaluated configurations. An important characteristic that differentiates between star and tree topologies is that in the latter, simultaneous transmissions were defined, and thus the complexity increased, making it necessary to take into account the channels through which the data is transmitted. Both similar and adjacent channels need to be avoided to reduce collisions and internal interference.

In star network, no parallel transmissions were defined, so the larger the channel denylist, the higher the quality of the network, the more packets are delivered, as this approach allows each device to use only the best channel. However, in a tree network it is not possible to always guarantee that the devices use only the best channel, because the best channel of a link can also be the best channel of another link that transmits in parallel, which generates collisions and ends up underestimating the real quality of the channel.

For this reason, different channel offsets were assigned, so that nodes always use different channels in the same timeslot. The AB-TSCH protocol takes this issue into account, and uses frequency hopping with different channel offsets, in addition to receiving help from LQE to identify which channels are best suited for temporary use.

In addition to the AB-TSCH which uses a dual channel list, and given the imprecision about the real quality of a channel, a triple channel list approach was proposed. It allows channels with dubious quality to be used only in special cases, such as the large packet loss of the network in general, for example. The best channels are always used until they have reduced quality and are denylisted or greylisted.

To prevent multiple channels from being blocked, thereby reducing diversity, a maximum number of four channels was established for the denylist. When there are more channels with low quality, the oldest ones inserted in the denylist are removed and inserted into the greyist, and the newer ones inserted in the denylist, in the same way as a FIFO (First In First Out) approach.

In this thesis, two and three channel lists were studied. The approach with three channel lists showed a superior performance, however this increases the network complexity and requires an additional field in the data frame. It is possible that additional lists increase the accuracy of the model, but it important to keep in mind that the complexity and the overhead also increase, which can compromise the throughput and energy consumption.

6.2 List of publications

6.2.1 Conference papers

- Diego V. Queiroz, Ruan D. Gomes, Cesar Benavente-Peces. Performance Evaluation of Default Active Message Layer (AM) and TKN15.4 Protocol Stack in TinyOS 2.1.2. In Proceedings of the 6th International Conference on Sensor Networks - Volume 1: SENSORNETS, 69-79, February/2017, Porto, Portugal. ISBN: 978-989-758-211-0, DOI: 10.5220/0006204200690079

- Ruan Delgado Gomes, Marcelo Alencar, Diego Véras de Queiroz, Iguatemi Eduardo da Fonseca, César Benavente-Peces. Comparison between Channel Hopping and Channel Adaptation for Industrial Wireless Sensor Networks. In Proceedings of the International Conference on Sensor Networks, At: Porto, Portugal, February/2017. DOI: 10.5220/0006206800870098

- Marcelo Alencar, César Benavente-Peces, Iguatemi Fonseca, Diego Véras de Queiroz, Ruan Delgado Gomes. Avaliação do Uso de Adaptação de Canal e Estimação de Qualidade de Enlace em Redes TSCH, XXXVI Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, December 2017, DOI: 10.14209/sbrt.2018.190.

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Evaluation of Channels Blacklists in TSCH Networks with Star and Tree Topologies. Q2SWinet'18: Proceedings of the 14th ACM International Symposium on QoS and Security for Wireless and Mobile Networks, October/2018, Pages 116–123, DOI: 10.1145/3267129.3267131.

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Join Multiple Channels and IEEE 802.15.4e TSCH Protocol Use Effects on WSN Performance and Energy Efficiency, 9th International Conference on Pervasive and Embedded Computing and Communication Systems, December 2018, DOI: 10.5220/0008162300830090.

- Gomes, R. D., Queiroz, D. V., Fonseca, I. E., Alencar, M. S., Benavente-Peces, C. Short course: Redes de Sensores sem Fio Industriais. In: XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2017, São Pedro, SP (short course proposal accepted).

6.2.2 Journal papers

- Diego Véras de Queiroz, Marcelo Sampaio de Alencar, Ruan Delgado Gomes, Iguatemi Eduardo da Fonseca, César Benavente-Peces. Survey and Systematic Mapping of Industrial Wireless Sensor Networks, August 2017. Journal of Network and Computer Applications. DOI: 10.1016/j.jnca.2017.08.019. Qualis 2016 ENGENHARIAS IV: A1

- Queiroz D.V., Benavente-Peces C., Gomes R.D. (2019) TinyOS-Based WSN Performance: Default Active Message Layer vs. TKN15.4. In: Benavente-Peces C., Cam-Winget N., Fleury E., Ahrens A. (eds) Sensor Networks. Communications in Computer and Information Science, vol 1074. Springer, Cham. DOI: 10.1007/978-3-030-30110-1_1. Qualis 2016 ENGENHARIAS IV: B5

- Gomes, R., Queiroz, D., Fonseca, I., & Alencar, M. (21-05-2017). A Simulation Model for Industrial Multi-Channel Wireless Sensor Networks. Journal of Communication and Information Systems, 32(1). DOI: 10.14209/jcis.2017.4. Qualis 2016 ENGENHARIAS IV: B1

- Ruan Delgado Gomes, Diego Véras de Queiroz, Abel Cavalcante Lima Filho, Marcelo Sampaio de Alencar. Real-Time Link Quality Estimation for Industrial Wireless Sensor Networks Using Dedicated Nodes. Ad Hoc Networks, Volume 59, 1 May 2017, Pages 116-133, Ad Hoc Networks. DOI: 10.1016/j.adhoc.2017.02.007. Qualis 2016 ENGENHARIAS IV: A1

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Channel Assignment in

TSCH-based Wireless Sensor Networks using Fuzzy Logic. **Submitted to Journal of Ambient Intelligence and Humanized Computing (2020). Qualis 2016 CIÊNCIA DA COMPUTAÇÃO: B1**

6.2.3 Book chapter

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. In book: Nanosensors for Smart Cities - Micro and Nano Technologies, 2020, Pages 515-526. DOI: 10.1016/B978-0-12-819870-4.00029-3

6.3 Future Research Directions

The two proposed solutions showed superior performance in terms of packet delivery and determinism, and as future work, they will be evaluated in experimental results with real devices in different types of industrial environments and for different types of applications, with different packet priorities.

As the network in tree topology adds more complexity to the solution, not only because of the simultaneous transmissions, but also due to scalability, since the tree network allows a larger expansion of the network than the star topology, this last topology will not be evaluated in the experiments.

The mesh topology, also proposed by the TSCH standard, adds even more complexity, as it allows larger network expansion than the tree topology, and also allows redundant routes (although the tree network allows it, but it is less common than the mesh network). Redundant routes using channel lists, with different channel offsets makes the task even more challenging, and they will be evaluated in future works.

The use of more accurate estimators will also be evaluated, involving not only the analysis of RSSI, the variation of RSSI and duplicate packages. These estimators will be classified using the approach proposed in the thesis, using fuzzy logic.

Other future work will be the development of algorithms for automatic network planning, that defines which topology to use and how many timeslots to assign for each sensor node. The use of multiple transceivers on the sink node will also be evaluated to reduce latency, thus allowing the cluster heads to transfer packets to the sink node simultaneously.

References

[Agrawal et al. 2014]AGRAWAL, P. et al. Long Term Channel Characterization for Energy Efficient Transmission in Industrial Environments. *IEEE Transactions on Communications*, v. 62, n. 8, p. 3004–3014, aug 2014. ISSN 0090-6778.

[Akerberg, Gidlund and Bjorkman 2011]AKERBERG, J.; GIDLUND, M.; BJORKMAN, M. Future research challenges in wireless sensor and actuator networks targeting industrial automation. In: *2011 9th IEEE International Conference on Industrial Informatics*. [S.l.]: IEEE, 2011. p. 410–415. ISBN 978-1-4577-0435-2.

[Alderisi et al. 2015]ALDERISI, G. et al. Simulative assessments of the IEEE 802.15.4e DSME and TSCH in realistic process automation scenarios. In: *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*. [S.l.]: IEEE, 2015. p. 948–955. ISBN 978-1-4799-6649-3.

[Algora et al. 2016]ALGORA, C. M. G. et al. Poster: Comparative study of EM-MAC and TSCH/orchestra for IoT. In: *2016 Symposium on Communications and Vehicular Technologies (SCVT)*. [S.l.]: IEEE, 2016. p. 1–6. ISBN 978-1-5090-4361-3.

[Amjad et al. 2016]AMJAD, M. et al. TinyOS-New Trends, Comparative Views, and Supported Sensing Applications: A Review. *IEEE Sensors Journal*, v. 16, n. 9, p. 2865–2889, may 2016. ISSN 1530-437X.

[Amzucu, Li and Fledderus 2014]AMZUCU, D. M.; LI, H.; FLEDDERUS, E. Indoor Radio Propagation and Interference in 2.4 GHz Wireless Sensor Networks: Measurements and Analysis. *Wireless Personal Communications*, v. 76, n. 2, p. 245–269, may 2014. ISSN 0929-6212.

[Anwar, Xia and Zhan 2015]ANWAR, M.; XIA, Y.; ZHAN, Y. TDMA based IEEE 802.15.4 for Low Latency Deterministic Control Applications. *IEEE Transactions on Industrial Informatics*, p. 1–1, 2015. ISSN 1551-3203.

[Baccour et al. 2012]BACCOUR, N. et al. Radio link quality estimation in wireless sensor networks. *ACM Transactions on Sensor Networks*, v. 8, n. 4, p. 1–33, sep 2012. ISSN 15504859.

[Baccour et al. 2015]BACCOUR, N. et al. Reliable link quality estimation in low-power wireless networks and its impact on tree-routing. *Ad Hoc Networks*, v. 27, p. 1–25, apr 2015. ISSN 15708705.

[Banik et al. 2018]BANIK, J. et al. SmartHop: A Cloud-Driven Channel Hopping Algorithm for Improved IoT Network Connectivity and Stability. In: *2018 IEEE International Conference on Communications (ICC)*. [S.l.]: IEEE, 2018. p. 1–6. ISBN 978-1-5386-3180-5.

[Barac et al. 2014]BARAC, F. et al. Channel Diagnostics for Wireless Sensor Networks in Harsh Industrial Environments. *IEEE Sensors Journal*, v. 14, n. 11, p. 3983–3995, nov 2014. ISSN 1530-437X.

[Baronti et al. 2007]BARONTI, P. et al. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. *Computer Communications*, v. 30, n. 7, p. 1655–1695, may 2007. ISSN 01403664.

[Barrenetxea et al. 2008]BARRENETXEA, G. et al. The hitchhiker's guide to successful wireless sensor network deployments. In: *Proceedings of the 6th ACM conference on Embedded network sensor systems - SenSys '08*. New York, New York, USA: ACM Press, 2008. p. 43. ISBN 9781595939906.

[Basmer, Schomann and Peter 2011]BASMER, T.; SCHOMANN, H.; PETER, S. Implementation analysis of the IEEE 802.15.4 MAC for wireless sensor networks. In: *2011 International Conference on Selected Topics in Mobile and Wireless Networking (iCOST)*. [S.l.]: IEEE, 2011. p. 7–12. ISBN 978-1-4577-2002-4.

[Bertocco, Gamba and Sona 2008]BERTOCCO, M.; GAMBA, G.; SONA, A. Is CSMA/CA really efficient against interference in a wireless control system? An experimental answer. In: *2008 IEEE International Conference on Emerging Technologies and Factory Automation*. [S.l.]: IEEE, 2008. p. 885–892. ISBN 978-1-4244-1505-2.

[Boccadoro et al. 2018]BOCCADORO, P. et al. Experimental Comparison of Industrial Internet of Things Protocol Stacks in Time Slotted Channel Hopping Scenarios. In: *2018 IEEE International Conference on Communications (ICC)*. [S.l.]: IEEE, 2018. p. 1–6. ISBN 978-1-5386-3180-5.

[Cheffena 2012]CHEFFENA, M. Industrial wireless sensor networks: channel modeling and performance evaluation. *EURASIP Journal on Wireless Communications and Networking*, v. 2012, n. 1, p. 1–8, 2012. ISSN 1687-1499.

[Chiti, Fantacci and Tani 2015]CHITI, F.; FANTACCI, R.; TANI, A. Performance Evaluation of an Adaptive Channel Allocation Technique for Cognitive Wireless Sensor Networks. In: *2015 IEEE Globecom Workshops (GC Wkshps)*. [S.l.]: IEEE, 2015. p. 1–6. ISBN 978-1-4673-9526-7.

[D. Gomes et al. 2017]D. Gomes, R. et al. Comparison between Channel Hopping and Channel Adaptation for Industrial Wireless Sensor Networks. In: *Proceedings of the 6th International Conference on Sensor Networks*. [S.l.]: SCITEPRESS - Science and Technology Publications, 2017. p. 87–98. ISBN 978-989-758-211-0.

[Dakdouk et al. 2018]DAKDOUK, H. et al. Reinforcement Learning Techniques for Optimized Channel Hopping in IEEE 802.15.4-TSCH Networks. In: *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems - MSWIM '18*. New York, New York, USA: ACM Press, 2018. p. 99–107. ISBN 9781450359603.

[Dargie and Poellabauer 2010]DARGIE, W.; POELLABAUER, C. *Fundamentals of Wireless Sensor Networks*. [S.l.]: John Wiley & Sons, Ltd, 2010. ISBN 9780470666388.

[De Couto et al. 2003]De Couto, D. S. J. et al. A high-throughput path metric for multi-hop wireless routing. In: *Proceedings of the 9th annual international conference on Mobile computing and networking - MobiCom '03*. New York, New York, USA: ACM Press, 2003. p. 134. ISBN 1581137532.

[De Guglielmo, Brienza and Anastasi 2016]De Guglielmo, D.; BRIENZA, S.; ANASTASI, G. A Model-based Beacon Scheduling algorithm for IEEE 802.15.4e TSCH networks. In: *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. [S.l.]: IEEE, 2016. p. 1–9. ISBN 978-1-5090-2185-7.

[Delgado Gomes et al. 2013]Delgado Gomes, R. et al. On the Impact of Local Processing for Motor Monitoring Systems in Industrial Environments Using Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, v. 2013, p. 1–14, 2013. ISSN 1550-1329.

[Du and Roussos 2013]DU, P.; ROUSSOS, G. Spectrum-aware wireless sensor networks. In: *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. [S.l.: s.n.], 2013. p. 2321–2325. ISSN 2166-9570.

[Du, Navarro and Mieyeville 2015]DU, W.; NAVARRO, D.; MIEYEVILLE, F. Performance evaluation of IEEE 802.15.4 sensor networks in industrial applications. *International Journal of Communication Systems*, v. 28, n. 10, p. 1657–1674, jul 2015. ISSN 10745351.

[Dunkels 2011]DUNKELS, A. The ContikiMAC Radio Duty Cycling Protocol. *SICS Technical Report T2011:13*, 2011. Available in: <http://soda.swedish-ict.se/5128/1/contikimac-report.pdf.>.

[Duquennoy et al. 2015]DUQUENNOY, S. et al. Orchestra. In: *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems - SenSys '15*. New York, New York, USA: ACM Press, 2015. p. 337–350. ISBN 9781450336314.

[Elsts et al. 2017]ELSTS, A. et al. Adaptive channel selection in IEEE 802.15.4 TSCH networks. In: *2017 Global Internet of Things Summit (GIoTS)*. [S.l.]: IEEE, 2017. p. 1–6. ISBN 978-1-5090-5873-0.

[Eriksson et al. 2009]ERIKSSON, J. et al. COOJA/MSPSim: interoperability testing for wireless sensor networks. In: *Proceedings of the Second International ICST Conference on Simulation Tools and Techniques*. [S.l.]: ICST, 2009. ISBN 978-963-9799-45-5.

[Eskola and Heikkilä 2016]ESKOLA, M.; HEIKKILÄ, T. Classification of Radio Channel disturbances for industrial wireless sensor networks. *Ad Hoc Networks*, v. 42, p. 19–33, may 2016. ISSN 15708705.

[Fonseca et al. 2007]FONSECA, R. et al. Four-bit wireless link estimation. In: *Proceedings of the Sixth ACM Workshop on Hot Topics in Networks (HotNets-VI)*. [S.l.: s.n.], 2007.

[Gnawali et al.]GNAWALI, O. et al. Interaction of retransmission, blacklisting, and routing metrics for reliability in sensor network routing. In: *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004*. [S.l.]: IEEE. p. 34–43. ISBN 0-7803-8796-1.

[Gomes, Watteyne and Krishnamachari 2017]GOMES, P. H.; WATTEYNE, T.; KRISHNAMACHARI, B. MABO-TSCH: Multihop and blacklist-based optimized time synchronized channel hopping. *Transactions on Emerging Telecommunications Technologies*, p. e3223, aug 2017. ISSN 21613915.

[Gomes et al. 2019]GOMES, R. D. et al. Adaptive and Beacon-based multi-channel protocol for Industrial Wireless Sensor Networks. *Journal of Network and Computer Applications*, v. 132, p. 22–39, apr 2019. ISSN 10848045.

[Gomes et al. 2017]GOMES, R. D. et al. A Simulation Model for Industrial Multi-Channel Wireless Sensor Networks. *Journal of Communication and Information Systems*, v. 32, n. 1, p. 29–40, 2017. ISSN 19806604.

[Gomes et al. 2017]GOMES, R. D. et al. Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes. *Ad Hoc Networks*, feb 2017. ISSN 15708705.

[Gunatilaka, Sha and Lu 2017]GUNATILAKA, D.; SHA, M.; LU, C. Impacts of channel selection on industrial wireless sensor-actuator networks. In: *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. [S.l.]: IEEE, 2017. p. 1–9. ISBN 978-1-5090-5336-0.

[Gungor and Korkmaz 2012]GUNGOR, V. C.; KORKMAZ, M. K. Wireless Link-Quality Estimation in Smart Grid Environments. *International Journal of Distributed Sensor Networks*, v. 8, n. 2, p. 214068, feb 2012. ISSN 1550-1477.

[Gursu et al. 2016]GURSU, M. et al. Reliable co-existence of 802.15.4e TSCH-based WSN and Wi-Fi in an aircraft cabin. In: *2016 IEEE International Conference on Communications Workshops (ICC)*. [S.l.]: IEEE, 2016. p. 663–668. ISBN 978-1-5090-0448-5.

[Hahm et al. 2015]HAHM, O. et al. Operating Systems for Low-End Devices in the Internet of Things: a Survey. *IEEE Internet of Things Journal*, 2015. ISSN 2327-4662.

[Hairong Yan et al. 2014]Hairong Yan et al. Superframe Planning and Access Latency of Slotted MAC for Industrial WSN in IoT Environment. *IEEE Transactions on Industrial Informatics*, v. 10, n. 2, p. 1242–1251, may 2014. ISSN 1551-3203.

[Hänninen et al. 2011]HÄNNINEN, M. et al. Link Quality-Based Channel Selection for Resource Constrained WSNs. In: . [S.l.: s.n.], 2011. p. 254–263.

[Huang, Pang and Hung 2009]HUANG, Y.-K.; PANG, A.-C.; HUNG, H.-N. A comprehensive analysis of low-power operation for beacon-enabled IEEE 802.15.4 wireless networks. *IEEE Transactions on Wireless Communications*, v. 8, n. 11, p. 5601–5611, nov 2009. ISSN 1536-1276.

[Incel 2011]INCEL, O. D. A survey on multi-channel communication in wireless sensor networks. *Computer Networks*, v. 55, n. 13, p. 3081–3099, sep 2011. ISSN 13891286.

[Jeong and Lee 2012]JEONG, W.-C.; LEE, J. Performance evaluation of ieee 802.15.4e dsme mac protocol for wireless sensor networks. In: *Enabling Technologies for Smartphone and Internet of Things (ETSIoT), 2012 First IEEE Workshop on*. [S.l.: s.n.], 2012. p. 7–12.

[Juc et al. 2016]JUC, I. et al. Energy consumption and performance of ieee 802.15.4e tsch and dsme. In: *2016 IEEE Wireless Communications and Networking Conference*. [S.l.: s.n.], 2016. p. 1–7.

[Kazmierkowski 2014]KAZMIERKOWSKI, M. P. Industrial Communication Technology Handbook, 2nd Ed. [Book News]. *IEEE Industrial Electronics Magazine*, v. 8, n. 2, p. 67–68, jun 2014. ISSN 1932-4529.

[Khanafer, Guennoun and Mouftah 2014]KHANAFER, M.; GUENNOUN, M.; MOUFTAH, H. T. A Survey of Beacon-Enabled IEEE 802.15.4 MAC Protocols in Wireless Sensor Networks. *IEEE Communications Surveys & Tutorials*, v. 16, n. 2, p. 856–876, jan 2014. ISSN 1553-877X.

[Kim, Shin and Cha 2008]KIM, Y.; SHIN, H.; CHA, H. Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks. In: *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)*. [S.l.]: IEEE, 2008. p. 53–63. ISBN 978-0-7695-3157-1.

[Kotsiou et al. 2017]KOTSIOU, V. et al. LABeL: Link-based Adaptive BLacklisting Technique for 6TiSCH Wireless Industrial Networks. In: *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems - MSWiM '17*. New York, New York, USA: ACM Press, 2017. p. 25–33. ISBN 9781450351621.

[Kotsiou et al. 2019]KOTSIOU, V. et al. Whitelisting without Collisions for Centralized Scheduling in Wireless Industrial Networks. *IEEE Internet of Things Journal*, p. 1–1, 2019. ISSN 2327-4662.

[Kotsiou et al. 2019]KOTSIOU, V. et al. Blacklisting-Based Channel Hopping Approaches in Low-Power and Lossy Networks. *IEEE Communications Magazine*, v. 57, n. 2, p. 48–53, feb 2019. ISSN 0163-6804.

[Kumar S., Ovsthus and Kristensen. 2014]Kumar S., A. A.; OVSTHUS, K.; KRISTENSEN., L. M. An Industrial Perspective on Wireless Sensor Networks - A Survey of Requirements, Protocols, and Challenges. *IEEE Communications Surveys & Tutorials*, v. 16, n. 3, p. 1391–1412, jan 2014. ISSN 1553-877X.

[Kusy et al. 2011]Kusy, B. et al. Radio diversity for reliable communication in wsns. In: *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*. [S.l.: s.n.], 2011. p. 270–281.

[Lee and Jeong 2012]LEE, J.; JEONG, W. C. Performance analysis of ieee 802.15.4e dsme mac protocol under wlan interference. In: *2012 International Conference on ICT Convergence (ICTC)*. [S.l.: s.n.], 2012. p. 741–746. ISSN 2162-1233.

[Mathew and Manuel 2015]MATHEW, V.; MANUEL, E. M. Cognitive scheduling in TSCH based mobile WSN using wavelet packet analysis. In: *2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*. [S.l.]: IEEE, 2015. p. 1–5. ISBN 978-1-4799-1823-2.

[Patti, Alderisi and Bello 2014]PATTI, G.; ALDERISI, G.; BELLO, L. L. Introducing multi-level communication in the IEEE 802.15.4e protocol: The MultiChannel-LLDN. In: *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*. [S.l.]: IEEE, 2014. p. 1–8. ISBN 978-1-4799-4845-1.

[Patti and Lo Bello 2016]PATTI, G.; Lo Bello, L. A Priority-Aware Multichannel Adaptive Framework for the IEEE 802.15.4e-LLDN. *IEEE Transactions on Industrial Electronics*, v. 63, n. 10, p. 6360–6370, oct 2016. ISSN 0278-0046.

[Peishuo Li et al. 2015]Peishuo Li et al. An adaptive channel selection scheme for reliable TSCH-based communication. In: *2015 International Symposium on Wireless Communication Systems (ISWCS)*. [S.l.]: IEEE, 2015. p. 511–515. ISBN 978-1-4673-6540-6.

[Peng Du and Roussos 2013]Peng Du; ROUSSOS, G. Spectrum-aware wireless sensor networks. In: *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. [S.l.]: IEEE, 2013. p. 2321–2325. ISBN 978-1-4673-6235-1.

[Petersen and Carlsen 2009]PETERSEN, S.; CARLSEN, S. Performance evaluation of WirelessHART for factory automation. In: *2009 IEEE Conference on Emerging Technologies & Factory Automation*. [S.l.]: IEEE, 2009. p. 1–9. ISBN 978-1-4244-2727-7.

[Queiroz et al. 2017]QUEIROZ, D. V. et al. Survey and systematic mapping of industrial Wireless Sensor Networks. *Journal of Network and Computer Applications*, v. 97, p. 96–125, nov 2017. ISSN 10848045.

[Queiroz, Gomes and Benavente-Peces 2017]QUEIROZ, D. V.; GOMES, R. D.; BENAVENTE-PECES, C. Performance Evaluation of Default Active Message Layer (AM) and TKN15.4 Protocol Stack in TinyOS 2.1.2. In: *Proceedings of the 6th International Conference on Sensor Networks*. [S.l.]: SCITEPRESS - Science and Technology Publications, 2017. p. 69–79. ISBN 978-989-758-211-0.

[Queiroz et al. 2018]QUEIROZ, D. V. et al. Evaluation of Channels Blacklists in TSCH Networks with Star and Tree Topologies. In: *Proceedings of the 14th ACM International Symposium on QoS and Security for Wireless and Mobile Networks - Q2SWinet'18*. New York, New York, USA: ACM Press, 2018. p. 116–123. ISBN 9781450359634.

[Rashvand and Abedi 2017]RASHVAND, H. F.; ABEDI, A. (Ed.). *Wireless Sensor Systems for Extreme Environments*. Chichester, UK: John Wiley & Sons, Ltd, 2017. ISBN 9781119126492.

[Rekik et al. 2015]REKIK, S. et al. Low-Power link quality estimation in smart grid environments. In: *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*. [S.l.]: IEEE, 2015. p. 1211–1216. ISBN 978-1-4799-5344-8.

[Rekik et al. 2016]REKIK, S. et al. Holistic link quality estimation-based routing metric for RPL networks in smart grids. In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. [S.l.]: IEEE, 2016. p. 1–6. ISBN 978-1-5090-3254-9.

[Sahoo, Pattanaik and Wu 2017]SAHOO, P. K.; PATTANAIK, S. R.; WU, S.-L. A novel ieee 802.15.4e dsme mac for wireless sensor networks. *Sensors*, v. 17, n. 1, 2017. ISSN 1424-8220. Available in: <http://www.mdpi.com/1424-8220/17/1/168>.

[Saifullah et al. 2014]SAIFULLAH, A. et al. Distributed Channel Allocation Protocols for Wireless Sensor Networks. *IEEE Transactions on Parallel and Distributed Systems*, v. 25, n. 9, p. 2264–2274, sep 2014. ISSN 1045-9219.

[Sha, Hackmann and Lu 2011]SHA, M.; HACKMANN, G.; LU, C. ARCH: Practical Channel Hopping for Reliable Home-Area Sensor Networks. In: *2011 17th IEEE Real-Time and Embedded Technology and Applications Symposium*. [S.l.]: IEEE, 2011. p. 305–315. ISBN 978-1-61284-326-1.

[Sohraby, Minoli and Znati 2007]SOHRABY, K.; MINOLI, D.; ZNATI, T. *Wireless Sensor Networks: Technology, Protocols, and Applications*. [S.l.]: Wiley, 2007. ISBN 978-0-471-74300-2.

[Soua and Minet 2015]SOUA, R.; MINET, P. Multichannel assignment protocols in wireless sensor networks: A comprehensive survey. *Pervasive and Mobile Computing*, v. 16, p. 2–21, jan 2015. ISSN 15741192.

[Tang et al. 2011]TANG, L. et al. EM-MAC: A dynamic multichannel energy-efficient MAC protocol for wireless sensor networks. In: *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing - MobiHoc '11*. New York, New York, USA: ACM Press, 2011. p. 1. ISBN 9781450307222.

[Tanghe et al. 2008]TANGHE, E. et al. The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 MHz. *IEEE Transactions on Wireless Communications*, v. 7, n. 7, p. 2740–2751, jul 2008. ISSN 1536-1276.

[Tavakoli et al. 2018]TAVAKOLI, R. et al. Dependable Interference-Aware Time-Slotted Channel Hopping for Wireless Sensor Networks. *ACM Transactions on Sensor Networks*, v. 14, n. 1, p. 1–35, jan 2018. ISSN 15504859.

[Tennina et al. 2013]TENNINA, S. et al. *IEEE 802.15.4 and ZigBee as Enabling Technologies for Low-Power Wireless Systems with Quality-of-Service Constraints*. [S.l.]: Springer Berlin Heidelberg, 2013. (SpringerBriefs in Electrical and Computer Engineering). ISBN 978-3-642-37367-1.

[Varga et al. 2016]VARGA, L.-O. et al. Why is frequency channel diversity so beneficial in wireless sensor networks? In: *2016 Wireless Days (WD)*. [S.l.]: IEEE, 2016. p. 1–3. ISBN 978-1-5090-2494-0.

[Wagner and Barton 2012]WAGNER, R. S.; BARTON, R. J. Performance comparison of wireless sensor network standard protocols in an aerospace environment: ISA100.11a and ZigBee Pro. In: *2012 IEEE Aerospace Conference*. [S.l.]: IEEE, 2012. p. 1–14. ISBN 978-1-4577-0557-1.

[Wang and Yao 2013]WANG, N.-C.; YAO, K. Characterizing fading channel under abrupt temporal variations. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. [S.l.]: IEEE, 2013. p. 5056–5060. ISBN 978-1-4799-0356-6.

[Watteyne et al. 2010]WATTEYNE, T. et al. Mitigating Multipath Fading through Channel Hopping in Wireless Sensor Networks. In: *2010 IEEE International Conference on Communications*. IEEE, 2010. p. 1–5. ISBN 978-1-4244-6402-9. Available in: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5502548>.

[Watteyne et al. 2012]WATTEYNE, T. et al. OpenWSN: a standards-based low-power wireless development environment. *Transactions on Emerging Telecommunications Technologies*, v. 23, n. 5, p. 480–493, aug 2012. ISSN 21613915.

[Wehrle, Günes and Gross 2010]WEHRLE, K.; GÜNES, M.; GROSS, J. *Modeling and Tools for Network Simulation*. [S.l.]: Springer Berlin Heidelberg, 2010. ISBN 978-3-642-12330-6.

[Will, Schleiser and Schiller 2009]WILL, H.; SCHLEISER, K.; SCHILLER, J. A real-time kernel for wireless sensor networks employed in rescue scenarios. In: *2009 IEEE 34th Conference on Local Computer Networks*. [S.l.]: IEEE, 2009. p. 834–841. ISBN 978-1-4244-4488-5.

[Wu et al. 2008]WU, Y. et al. Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks. In: *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*. [S.l.]: IEEE, 2008. p. 1193–1201. ISBN 978-1-4244-2026-1.

[Zadeh 1965]ZADEH, L. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, jun 1965. ISSN 00199958.

[Zhan, Xia and Anwar 2016]ZHAN, Y.; XIA, Y.; ANWAR, M. GTS size adaptation algorithm for IEEE 802.15.4 wireless networks. *Ad Hoc Networks*, v. 37, p. 486–498, feb 2016. ISSN 15708705.

[Zhou et al. 2006]ZHOU, G. et al. MMSN: Multi-Frequency Media Access Control for Wireless Sensor Networks. In: *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*. [S.l.]: IEEE, 2006. p. 1–13. ISBN 1-4244-0221-2.

[Zorbas et al. 2018]ZORBAS, D. et al. LOST: Localized blacklisting aware scheduling algorithm for IEEE 802.15.4-TSCH networks. In: *2018 Wireless Days (WD)*. [S.l.]: IEEE, 2018. p. 110–115. ISBN 978-1-5386-5633-4.

[Zorbas, Papadopoulos and Douligeris 2018]ZORBAS, D.; PAPADOPOULOS, G. Z.; DOULIGERIS, C. Local or Global Radio Channel Blacklisting for IEEE 802.15.4-TSCH Networks? In: *2018 IEEE International Conference on Communications (ICC)*. [S.l.]: IEEE, 2018. p. 1–6. ISBN 978-1-5386-3180-5.

APPENDIX A – Fuzzy Logic Code in Python

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
\%matplotlib inline
import matplotlib.pyplot as plt

# New Antecedent/Consequent objects hold universe variables
# and membership functions

# Considering range from -40dBm, -70dBm e -100dBm and taking account
# -70dBm as default, -40dBm is 42% better than the default and
# -100dBm is 42% worse than the default
prr = ctrl.Antecedent(np.arange(0, 101, 1), 'PDR (%)') # 0 to 100%
rssi = ctrl.Antecedent(np.arange(-42, 43, 1), 'Variarion of RSSI (%)')
        # 0 to 100%
duplicate = ctrl.Antecedent(np.arange(0, 101, 1), 'Duplicate (%)')
        # 0 to 100%
channel_list = ctrl.Consequent(np.arange(0, 101, 1), 'Channel lists')
        # 0 to 100%

# Automatically creates mapping between crisp and fuzzy values
# using a standard membership function (triangle)
prr['bad'] = fuzz.gaussmf(prr.universe, 0, 18)
prr['acceptable'] = fuzz.gaussmf(prr.universe, 65, 8)
prr['great'] = fuzz.gaussmf(prr.universe, 100,10)

# Creates membership functions using different types
rssi['bad'] = fuzz.gaussmf(rssi.universe, -42, 18)
rssi['acceptable'] = fuzz.gaussmf(rssi.universe, 0, 5)
rssi['great'] = fuzz.gaussmf(rssi.universe, 42,18)
```

```
# Automatically creates mapping between crisp and fuzzy values
# using a standard membership function (triangle)
duplicate['suitable'] = fuzz.gaussmf(duplicate.universe, 0, 18)
duplicate['acceptable'] = fuzz.gaussmf(duplicate.universe, 40, 8)
duplicate['bad'] = fuzz.gaussmf(duplicate.universe, 100,25)

channel_list['Denylist'] = fuzz.gaussmf(prr.universe, 0, 16)
channel_list['Greylist'] = fuzz.gaussmf(prr.universe, 50, 8)
channel_list['Allowlist'] = fuzz.gaussmf(prr.universe, 100,16)

rule1 = ctrl.Rule(prr['great'] & (rssi['great'] | rssi['acceptable']),
        channel_list['Allowlist'])
rule2 = ctrl.Rule(prr['bad'] | rssi['bad'], channel_list['Denylist'])
rule3 = ctrl.Rule(prr['bad'] & rssi['acceptable'],
        channel_list['Denylist'])
rule4 = ctrl.Rule((prr['great'] | prr['acceptable']) &
        (rssi['bad'] | rssi['acceptable']), channel_list['Greylist'])
rule5 = ctrl.Rule(prr['acceptable'] & (rssi['acceptable'] |
        rssi['great']), channel_list['Allowlist'])
rule6 = ctrl.Rule(((prr['acceptable'] | prr['great'] | prr['bad']) &
        duplicate['bad']), channel_list['Denylist'])
rule7 = ctrl.Rule(duplicate['acceptable'] & (rssi['acceptable'] |
        rssi['great']), channel_list['Greylist'])
rule8 = ctrl.Rule((duplicate['suitable'] & prr['bad']),
        channel_list['Allowlist'])

lists_ctrl = ctrl.ControlSystem([rule1, rule2, rule3, rule4,
        rule5, rule6, rule7, rule8])
lists_simulator = ctrl.ControlSystemSimulation(lists_ctrl)

# Entering some values
lists_simulator.input['PDR (\%)'] = 95
lists_simulator.input['Variarion of RSSI (\%)'] = 3
lists_simulator.input['Duplicate (\%)'] = 15

# Computing the result
lists_simulator.compute()
print(lists_simulator.output['Channel lists'])
```

Universidad Politécnica de Madrid
Universidade Federal de Campina Grande
Doctorado en Ingeniería de Sistemas y Servicios para la Sociedad de la Información
Doutorado em Engenharia Elétrica
Redes, Sistemas, Servicios y Tecnologías de Telecomunicación

Contribuições para Redes de Sensores sem Fio Industriais (Resumo)

Diego Véras de Queiroz

Campina Grande, Brasil
10/2020

Diego Véras de Queiroz

Contribuições para Redes de Sensores sem Fio Industriais (Resumo)

Resumo da tese de doutorado apresentada à Coordenação do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande em um convênio de cotutela com a Universidade Politécnica de Madrid como requisito necessário para a obtenção do grau de Doutor em Ciências na área de Engenharia Elétrica / Ingeniería de Sistemas y Servicios para la Sociedad de la Información.

Orientadores:

Marcelo Sampaio de Alencar (UFCG) / Cesar Benavente-Peces (UPM)

Co-orientador:

Iguatemi Eduardo da Fonseca (UFPB)

UFCG – Universidade Federal de Campina Grande
UPM – Universidad Politécnica de Madrid
UFPB – Universidade Federal de Paraíba

Campina Grande, Brasil

10/2020

Contribuições para redes de sensores sem fio industriais (Resumo)

Autor: Diego Véras de Queiroz

Orientadores: Marcelo Sampaio de Alencar, Ph.D., Cesar Benavente-Peces, Ph.D.

Co-orientador: Iguatemi Eduardo da Fonseca, Ph.D.

Resumo

Avanços recentes nas redes de sensores sem fio (WSN), especialmente em ambientes industriais (IWSN), trouxeram melhorias importantes para implantação de uma rede de sensores nesses ambientes. No entanto, há um alto nível de interferência, ruído, sombreamento e desvanecimento por múltiplos percursos na indústria, causados por máquinas, objetos metálicos e obstruções, que podem afetar a qualidade do enlace. Alguns mecanismos foram propostos para mitigar os efeitos negativos, como salto de frequência e lista negra de canais. No entanto, ainda existem questões em aberto, como um método apropriado de gerenciar ambos os mecanismos para evitar canais similares ou adjacentes em transmissões simultâneas. O uso de um estimador de qualidade de enlace (LQE), com uma atribuição adequada de canais, ajuda a aumentar o desempenho da rede. Além do uso de estimadores, esta tese propõe duas abordagens que separam os canais de acordo com o perfil temporário de cada um e os insere em listas. O objetivo é lidar melhor com os efeitos negativos do ambiente no meio de transmissão. A primeira usa lista dupla de canais (adequados e inadequados), e a segunda utiliza uma lista tripla de canais (adequados, inadequados e cinza). Em ambas foi levado em consideração o gerenciamento do deslocamento do canal (*channel offset*), de maneira a evitar colisões e diminuir a interferência interna em transmissões simultâneas. A lista de canais inadequados apresenta canais de baixa qualidade, a lista cinza possui canais com qualidade incerta, e a lista de canais adequados possui canais com boa qualidade. Um método de lógica difusa foi utilizado na abordagem com lista tripla para classificar os canais na lista mais adequada. As duas propostas foram comparadas por meio de simulação utilizando o método de salto em frequência baseado no protocolo TSCH do padrão IEEE 802.15.4e para WSN. No estudo, foram analisadas redes nas topologias em estrela e em árvore, com e sem estimadores de qualidade, com e sem colisões, utilizando um modelo de canal realista para IWSN. Os resultados dos experimentos mostraram que na topologia em árvore as abordagens com lista dupla e tripla apresentaram um melhor desempenho do que a abordagem sem estimadores de qualidade, em termos de taxa de entrega de pacotes e determinismo, e que a lista tripla superou a lista dupla em redes mais dinâmicas. Na topologia em estrela, quanto maior o tamanho da lista de canais inadequados, melhor o desempenho da rede, pois os dispositivos utilizam apenas os melhores canais.

Palavras-chave: Redes de sensores sem fio industriais; salto em frequência; deslocamento de canal; lista de canais; lógica difusa.

Lista de Figuras

Lista de Tabelas

Lista de abreviações e acrônimos

6LoWPAN – IPv6 over Low power Wireless Personal Area Networks

AB-TSCH – Adaptive-Blacklist TSCH

ACK – Pacote de Reconhecimento

AM – Active Message Protocol

AMCA – Asynchronous Multi-Channel Adaptation

BI – Beacon Interval

BL – Channel Denylist

BLINK – Frequency Identification Blink

BSD – Berkeley Software Distribution

CAP – Período de Contenção de Acesso

CFP – Período Livre de Contenção

CH – Cluster Head

CoAP – Constrained Application Protocol

CSMA/CA – Carrier Sense Multiple Access with Collision Avoidance

DSME – Deterministic and Synchronous Multi-channel Extension

EB – Enhanced Beacons

EM-MAC - Energy-Efficient MAC Protocol

FIFO – First In, First Out

FIT/ IoT Lab – Large scale IoT testbed

GTS – Guaranteed Time Slots

IEEE – Institute of Electrical and Electronics Engineers

IETF – Internet Engineering Task Force

IIoT – Internet das Coisas Industriais

IoT – Internet das Coisas

IPv6 – Internet Protocol version 6

IWSN – Redes de Sensores sem Fio Industriais

LLDN – Low Latency Deterministic Network

LLN - Low Power and Lossy Networks

LOS – Line-Of-Sight

LQE – Link Quality Estimator

LQI – Link Quality Indicator

LR-WPAN – Low-rate, Wireless Personal Area Network

MAC - Media Access Control layer

MCU – Microcontrolador

NDLNC4 – No denylists, no collisions, four cluster heads

NS2/NS3 – Network Simulator, versions 2 and 3

OMNeT++ – Objective Modular Network Testbed

OS – Sistema Operacionai

PDR – Taxa de Entrega de Pacotes

QoS – Qualidade de Serviço

RMS – Root-Mean-Square

RNP – Required Number of Packet Transmissions

RPL – IPv6 Routing Protocol for LLNs

RSSF – Redes de Sensores sem Fio

RSSI – Received Signal Strength Indication

TDMA – Time Division Multiple Access

TG4e – Task Group 4e

TKN15.4 – Protocol Stack of IEEE 802.15.4 in TinyOS

TMCP – Tree-Based Multi-Channel Protocol

TOSSIM – Simulador TinyOS

TSCH – Time Slotted Channel Hopping

WL – Channel Allowlist

WSN – Wireless Sensor Networks

YDLYC4 – Com denylists, com colisões, quatro cluster heads

YTLNC4 – Com triple list, sem colisões, quatro cluster heads

YTLYC4 – Com triple list, com colisões, quatro cluster heads

Lista de Símbolos

n – Expoente de perda de percurso

d_0 – Distância de referência

$L(d_0)$ – Perda de percurso na distância de referência

X_σ – Desvio padrão do sombreamento

K – Fator Rice

K_σ – Desvio padrão do fator Rice

ASN – Número de timeslots da rede desde o início

FHS_{length} – Número de canais disponíveis

N_{ts} – Número de timeslots alocados aos coordenadores

SF – Duração do slotframe em milissegundos

T_{ts} – Duração do timeslot em milissegundos

D_{end} – Número de nós finais conectados a cada cluster head

N_{ch} – Número de cluster heads

Sumário

1 Introdução

Tradicionalmente, os sistemas de automação industrial são construídos utilizando meios de transmissão cabeados [Queiroz et al. 2017], e esses sistemas apresentam pouca flexibilidade, além de altos custos de instalação e manutenção. Para implantar uma rede de sensores com fio, uma infraestrutura de rede precisa ser implementada, o que pode gerar altos custos de adaptação da indústria e transposição de barreiras físicas para instalação dos cabos de transmissão. Além disso, caso seja necessário alterar a topologia da rede devido à posição dos nós sensores, a infraestrutura da rede precisa ser adaptada, o que também pode representar um custo elevado.

Uma alternativa para a construção de sistemas de monitoramento e controle flexíveis e de baixo custo com fácil instalação e manutenção é o uso de redes de sensores sem fio (RSSF). Algumas das principais preocupações ao projetar uma RSSF são a eficiência energética, confiabilidade e pontualidade nas comunicações, e as duas últimas são questões críticas em aplicações industriais e na área de saúde [De Guglielmo, Brienza and Anastasi 2016].

Avanços recentes na área de pesquisa de RSSF, especialmente em ambientes industriais (IWSN), trouxeram melhorias importantes na implantação de uma rede de sensores em tais ambientes. No entanto, um desafio ainda existente é a alta interferência e ruído causado por máquinas, por diversos objetos metálicos e obstruções, conforme ilustrado na Figura 1, que podem afetar a qualidade do enlace.

O uso de IWSN está sujeito a problemas típicos das comunicações sem fio, como ruído, sombreamento, desvanecimento por multipercurso e interferência. Além disso, o canal sem fio em muitos ambientes industriais não é estacionário quando considerados longos períodos, o que pode causar mudanças abruptas nas características do canal ao longo do tempo. Os canais utilizados para transmissão de dados podem apresentar temporariamente baixa qualidade ou mesmo impossibilitar a comunicação entre os sensores, devido aos problemas mencionados.

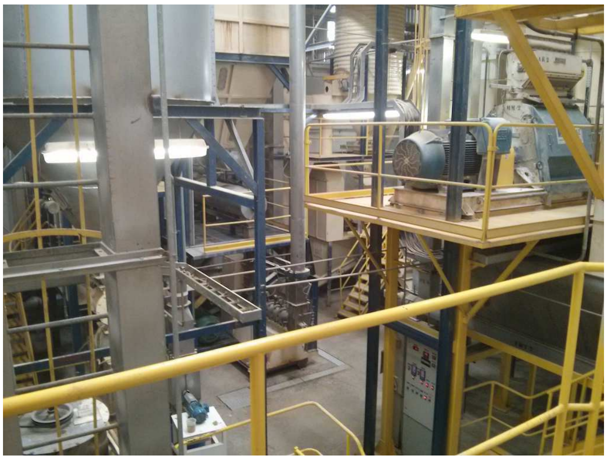

Figura 1: Ambiente Industrial.

Muitos sistemas de monitoramento industrial precisam processar sinais heterogêneos que se modificam rapidamente, e neste tipo de aplicação, a operação da rede deve ocorrer de forma determinística, uma vez que as informações monitoradas precisam ser entregues aos sistemas de controle periodicamente, acompanhando os tempos de entrega de cada pacote. Além disso, os nós sensores possuem restrições de recursos, apresentando baixa capacidade de processamento e limitações de bateria.

Outro problema que pode afetar o desempenho da rede é a assimetria do enlace, que é a diferença entre a qualidade do canal sem fio nas duas direções. Em RSSFs, a maioria dos pacotes são geralmente transmitidos em uma direção, do nó final para o nó receptor. No entanto, muitos protocolos usam reconhecimento de pacote (pacotes ACK) ou pacotes de controle e, nesses casos, é importante garantir que o enlace tenha boa qualidade em ambos os lados.

Apesar das desvantagens citadas, a alternativa mais viável para a construção de sistemas de monitoramento e controle em ambientes industriais é a utilização de redes sem fio, que são mais flexíveis, apresentam menor custo, são auto-organizadas, fáceis de instalar e manter, e possuem capacidade de processamento local.

Alguns mecanismos foram propostos para mitigar os efeitos negativos do ambiente nos canais sem fio, como salto de frequência e lista negra de canais. No entanto, ainda existem questões em aberto, como o método adequado de gerenciamento de ambos os mecanismos para evitar canais semelhantes ou adjacentes de nós vizinhos, o que pode levar a colisões e interferências no espectro utilizado pelos sensores.

1.1 Problema de Pesquisa

O problema de pesquisa está relacionado à dificuldade de utilização de redes de sensores na indústria devido à problema enfrentado pelos dispositivos de rede em relação à interferência e desvanecimento por múltiplos percursos. O desafio é alcançar uma alta taxa de entrega de pacotes (PDR) com baixo consumo de energia e baixo atraso nesses ambientes.

Nem todos os 16 canais disponíveis definidos pelo padrão IEEE 802.15.4 apresentam as mesmas características ao mesmo tempo, e alguns deles apresentam pelo menos qualidade aceitável, e devem ser usados pelos sensores, enquanto outros canais apresentam baixa qualidade e devem ser temporariamente bloqueados. Por outro lado, com menos canais disponíveis, a rede diminui sua diversidade e aumenta a probabilidade de colisões e interferência internas, principalmente quando a rede possui muitos dispositivos com transmissões simultâneas.

Apesar da menor diversidade e maior probabilidade de colisões, o uso de listas de canais tende a aumentar o PDR em ambientes mais conturbados e dinâmicos. Em ambientes menos problemáticos, geralmente de escritórios e residências, com menos mudanças na topologia da rede, o uso de listas de canais não parece ter efeitos positivos consideráveis sobre o PDR, como é apresentado ao longo da tese.

Assim, em ambientes industriais é necessário realizar a separação dos canais por meio de listas diferentes de acordo com o perfil de cada canal em um determinado momento para que possam ser gerenciados da melhor forma e tais problemas possam ser mitigados. O perfil temporário de cada canal em cada enlace em determinados momentos é avaliado por meio de estimadores de qualidade (LQE).

A ideia principal é propor um método que gerencie as listas de canais por meio de um LQE, e que esse gerenciamento considere as transmissões simultâneas para evitar interferências internas e colisões.

A Figura 2 apresenta o exemplo de uma aplicação com topologia em árvore usada na tese. Além da topologia em árvore, também foi avaliada a topologia em estrela, na qual existe um nó sorvedouro que atua como coordenador da rede e LQE. Na topologia em árvore, uma rede com quatro *Cluster Heads* (CH) e 48 nós finais conectados aos CH foi avaliada. Cada CH está diretamente conectado a 12 nós finais diferentes. Ele funciona como coordenador e roteador e encaminha os pacotes dos nós finais para o nó sorvedouro. Em uma topologia em estrela, 16 nós finais são usados para enviar pacotes de dados ao

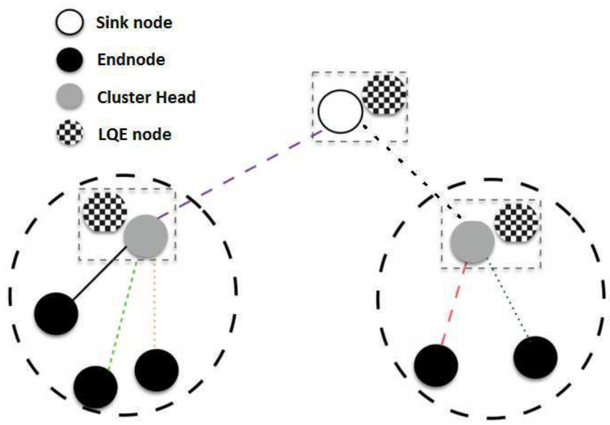

Figura 2: Exemplo de topologia usada na tese.

nó sorvedouro.

É uma tarefa desafiadora configurar as listas de canais e agendar as transmissões dos nós de forma que dois ou mais enlaces vizinhos com transmissões paralelas não usem os mesmos canais ou canais adjacentes ao mesmo tempo. Embora melhore o desempenho geral, o uso de listas de canais pode diminuir a capacidade da rede, uma vez que o mesmo tráfego deve ser encaminhado por um número menor de canais. Porém, devido à grande variação na qualidade dos enlaces, o uso de listas garante que os sensores não saltem para canais com baixa qualidade, permitindo um aumento no PDR e uma redução no consumo de energia devido à redução do número de retransmissões.

A importância do gerenciamento correto da lista de canais tem sido negligenciada pela maioria dos trabalhos, que apresentam mais estudos sobre políticas de escalonamento, eficiência energética e redes *multi-hop* [Queiroz et al. 2017], e realizam o que se chama de salto cego pelos canais, sem avaliação prévia da qualidade de cada um. Esta tese aborda essa questão e propõe uma solução que gerencia os canais de uma forma que considere as características locais do meio sem fio, e busca atribuir canais espectralmente distantes a dispositivos com transmissão simultânea.

É proposto o método de salto de frequência, baseado no padrão industrial IEEE 802.15.4e, mais especificamente, do modo TSCH do IEEE 802.15.4e, com atribuição de canais espectralmente distantes para nós com transmissões paralelas por meio de gerenciamento de deslocamentos de canal. Além disso, duas propostas são usadas para listas de canais: uma que usa duas listas de canais e outra que usa três listas de canais. Ambas as soluções apresentaram desempenho superior quando comparadas à abordagem de salto cego baseada no protocolo TSCH, com e sem listas de canais.

1.2 Principais Contribuições

- Foram realizados estudos experimentais para avaliação do canal sem fio em ambientes industriais;
- Os estudos por simulação foram realizados utilizando um modelo de canal desenvolvido que simula ambientes industriais, com o objetivo de avaliar os métodos de salto em frequência e adaptação de canal, identificando as vantagens e desvantagens de cada um;
- Também foram realizados estudos experimentais em ambiente de laboratório utilizando um único canal, com diversos canais por meio de salto em frequência e com uso de listas de canais inadequados;
- Foi proposto um novo protocolo de camada de enlace, que utiliza estimador de qualidade de enlace nos nós receptores. Ele realiza saltos de frequência e utiliza lista dupla de canal, que divide os canais de acordo com a qualidade de cada um;
- Foi proposto um método semelhante ao protocolo desenvolvido, porém este utiliza uma lista tripla de canais por meio de um conjunto de lógica *fuzzy*, além de utilizar estimadores de qualidade de enlace e salto de frequência. A lista tripla permite diferenciar melhor a qualidade de cada canal.

1.3 Esboço da Tese

O restante deste trabalho está organizado da seguinte forma.

- Capítulo 2 apresenta o estado da arte sobre as pesquisas de redes de sensores sem fio voltadas para ambientes industriais, incluindo as questões de pesquisa em aberto, e o modelo de canal desenvolvido para simular os ambientes industriais;
- Capítulo 3 apresenta uma visão geral do padrão IEEE 802.15.4e, especialmente do protocolo TSCH;
- Capítulo 4 propõe um novo protocolo de camada de enlace para IWSN baseado no modo TSCH denominado Adaptive-Blacklist TSCH (AB-TSCH) e apresenta os resultados de experimentos simulados com o modelo de canal desenvolvido;

- Capítulo 5 apresenta uma proposta de melhoria do método de lista de canais inadequados, usando outras duas listas de canais por um método de lógica *fuzzy* que ajuda a classificar os canais nestas listas. Os resultados também são apresentados e comparados com outras abordagens com e sem lista de canais inadequados;

- Finalmente, o Capítulo 6 apresenta algumas ideias de pesquisa para trabalhos futuros.

2 Redes industriais de sensores sem fio

Os nós de uma RSSF são equipados com sensores (ou atuadores), possuem capacidades de processamento, além de restrições de recursos com baixo poder de processamento e, em alguns casos, apresentam restrições quanto ao consumo de energia.

Na indústria, sensores são inseridos no ambiente para monitorar parâmetros críticos como vibração, temperatura, pressão e eficiência do motor [Delgado Gomes et al. 2013]. As medições por eles obtidas são transmitidas sem fio a um nó sorvedouro, que fornece as informações para análise por uma central de monitoramento, ou para serem utilizadas em sistemas de controle. Em alguns casos, os dados adquiridos são processados localmente (no nó final) antes da transmissão.

Uma variedade de informações pode ser adquirida pela RSSF com diferentes objetivos, o que permite uma tomada de decisão adequada. Com base nas informações obtidas, é possível consertar ou substituir o equipamento antes que ocorram maiores perdas.

O uso de RSSF em sistemas industriais apresenta alguns desafios. As redes sem fio usam um meio de comunicação inerentemente não confiável, que pode ser agravado devido ao ruído e à interferência na banda do espectro usada para as comunicações. Diferentes tipos de fontes de interferência para RSSF podem ser encontrados em ambientes industriais, como equipamentos de soldagem, fornos de microondas e outros dispositivos de comunicações sem fio (por exemplo, redes Wi-Fi ou Bluetooth), conforme representado na Figura 3. Além disso, em ambientes industriais existem máquinas e muitos objetos metálicos e obstruções que podem afetar a qualidade do canal sem fio. Assim, além de ruído e interferência, o canal é afetado por fortes efeitos de propagação por multipercurso [Cheffena 2012], o que resulta em um alto grau de atenuação em grande e pequena escala [Tanghe et al. 2008].

Em comparação com outros ambientes internos e externos, os ambientes industriais são mais severos devido às variações imprevisíveis de temperatura, pressão, umidade e

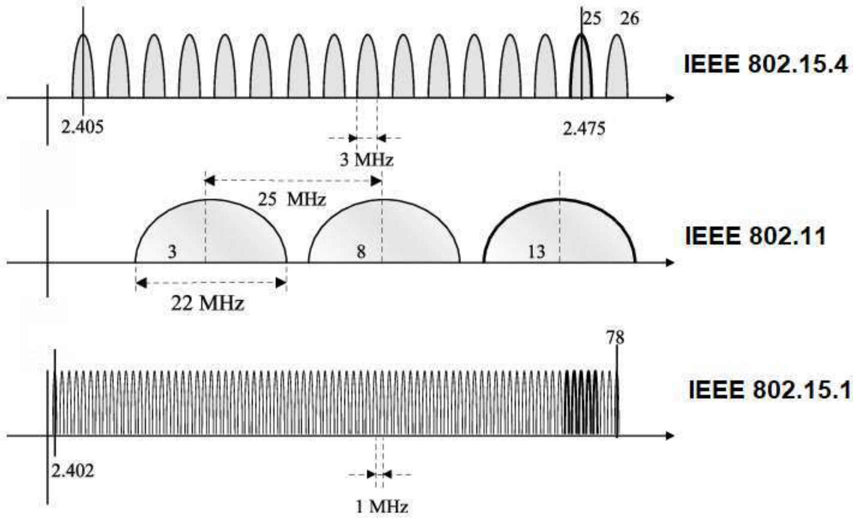

Figura 3: Tecnologias que podem interferir umas nas outras, adaptado de [Bertocco, Gamba and Sona 2008].

assim por diante. Além disso, o canal sem fio em muitas indústrias não é estacionário quando considerado um longo período de monitoramento, o que pode causar mudanças abruptas nas características do canal ao longo do tempo [Agrawal et al. 2014].

A Fig. 4 mostra uma configuração geral de uma RSSF, na qual as setas tracejadas representam os enlaces sem fio dos nós finais, e as setas sólidas representam os enlaces entre os nós de agrupamento (CH) e os nós sorvedouros. Os nós sorvedouros podem ter várias interfaces sem fio, cada uma usando um canal diferente para permitir transmissões simultâneas dos CHs. Os nós finais podem se comunicar com um nó sorvedouro usando um único salto ou vários saltos por meio de um CH ou roteadores intermediários (nós ilustrados em verde). Nesse cenário, os roteadores intermediários são nós que podem atuar tanto como dispositivo de sensoriamento (nó final) quanto como roteador. O Nó de Processamento Final (*Final Processing Node*) está conectado aos nós sorvedouros, e é responsável por coletar e analisar todos os dados recebidos da RSSF. Normalmente, a conexão entre os nós sorvedouro e o nó de processamento final é feita por meio de um enlace com fio.

Para lidar com os efeitos adversos inerentes ao meio sem fio na indústria, vários padrões foram desenvolvidos e estão ilustrados na Figura 5, em que o WirelessHART e o ISA100.11a são os mais conhecidos para dispositivos de capacidade reduzida, com o padrão IEEE 802.15.4e como o mais recente. O padrão ZigBee, que é baseado no IEEE 802.15.4-2003 da camada de controle de acesso ao meio (MAC), foi desenvolvido para RSSF de uso geral, e ainda é amplamente utilizado para pesquisas em ambientes industriais, embora esteja sendo gradualmente substituído por tecnologias mais apropriadas,

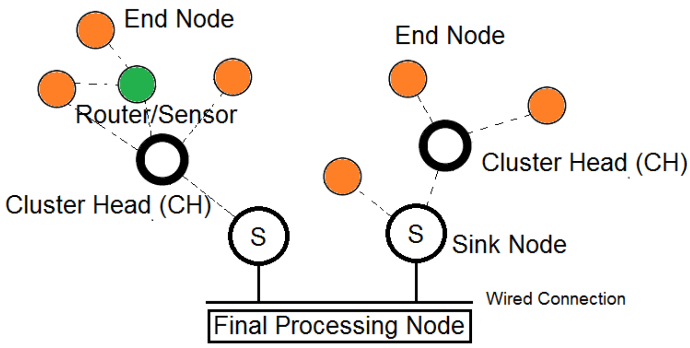

Figura 4: RSSF geral.

como o WirelessHART e o IEEE 802.15.4e [Queiroz, Gomes and Benavente-Peces 2017]. Lançado oficialmente em 2007, o WirelessHART continua sendo um dos principais padrões pesquisados para a indústria, e devido às melhorias envolvendo o padrão IEEE 802.15.4, o IEEE 802.15.4e tem chamado a atenção dos pesquisadores.

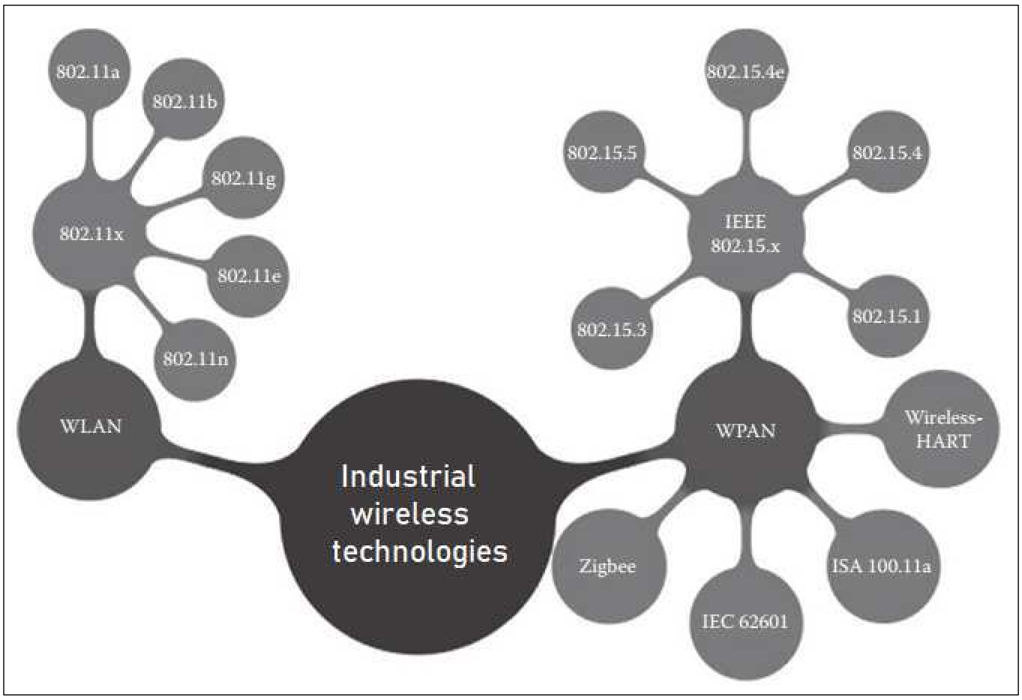

Figura 5: Tecnologias de rede sem fio industriais, adaptadas de [Kazmierkowski 2014].

O WirelessHART e ISA100.11a usam a camada física do IEEE 802.15.4, mas definem diferentes protocolos para a camada MAC baseados no método de acesso TDMA, salto de frequência ou uma combinação de ambos os mecanismos. Mesmo definindo mecanismos para lidar com a falta de confiabilidade, esses protocolos ainda podem enfrentar alguns problemas. Por exemplo, ao usar o salto de frequência, os nós geralmente mudam para um novo canal antes de cada transmissão. No entanto, se o gerenciamento adequado da lista

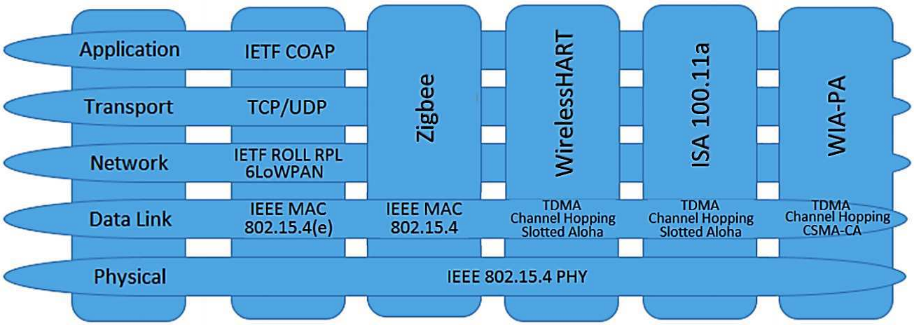

Figura 6: Padronização de tecnologias sem fio com baixo consumo de energia e recursos limitados.

de canais não for feito, o desempenho da rede pode ser significativamente degradado, já que o nó provavelmente irá pular para um canal com baixa qualidade [Gursu et al. 2016].

Conforme mencionado brevemente, para superar as limitações do padrão IEEE 802.15.4-2003, o IEEE criou o 802.15 Task Group 4e (TG4e), que redesenha os protocolos MAC 802.15.4 existentes. O objetivo era definir um protocolo MAC de salto de frequência de baixa potência, capaz de atender às necessidades das aplicações industriais. Essa melhoria resultou no documento IEEE 802.15.4e MAC, lançado em 2012. Esse padrão usa muitas ideias dos padrões WirelessHART e ISA-100.11a, incluindo acesso por timeslots compartilhados e dedicados e comunicações por meio de múltiplos canais. Especificamente, o IEEE 802.15.4e estende o padrão IEEE 802.15.4 anterior introduzindo cinco novos modos de operação. Entre eles, apenas TSCH, DSME e LLDN foram explorados na literatura, até o momento [Queiroz, Gomes and Benavente-Peces 2017].

2.1 Aspectos importantes no projeto de RSSF

Os principais aspectos a serem considerados no projeto de uma RSSF para ambientes industriais são o uso de múltiplos canais e o uso de estimadores de qualidade de enlace, que auxiliam os sensores a executarem eficientemente o método de salto de frequência.

2.1.1 Abordagem de múltiplos canais

Vários protocolos que implementam múltiplos canais foram desenvolvidos para RSSF, porém neste tópico sua classificação considera que apenas os nós coordenadores podem ter múltiplos transceptores. Assim, aos nós finais é atribuída a tarefa de apenas detectar e transmitir os dados, economizando energia e garantindo a estabilidade da rede. Os protocolos multicanais podem ser divididos em estáticos, dinâmicos e semidinâmicos [Soua and Minet 2015].

2.1.1.1 Protocolos Estáticos

Nos protocolos estáticos, não há mecanismo de sincronização ou sobrecarga em relação às mudanças de canal. Essas características os tornam os mais simples de implementar, em comparação com outras abordagens. A desvantagem é que não conseguem se adaptar às variações das características do canal ao longo do tempo, pois são configurados apenas no início da rede.

2.1.1.2 Protocolos Dinâmicos

Nesta categoria, os protocolos utilizam vários canais simultaneamente para reduzir as colisões por meio de estratégias de escalonamento. Essas estratégias evitam que o mesmo canal seja usado no mesmo timeslot por nós com transmissões simultâneas. Para isso, é necessário definir mecanismos de sincronização para garantir que o transmissor e o receptor estejam no mesmo canal durante a transmissão. A desvantagem surge quando os pacotes são transmitidos em broadcast e quando novos nós são adicionados [Incel 2011]. Os padrões WirelessHART, ISA100.11a e o TSCH utilizam a abordagem dinâmica

2.1.1.3 Protocolos Semi-dinâmicos

Nesta categoria de protocolos, a troca de canal e a transmissão simultânea em diferentes canais são permitidas, mas a troca ocorre com menos frequência. Em algumas abordagens, o nó receptor é associado a canais fixos e o transmissor muda para o canal de seu receptor quando há pacotes para transmitir. Nesse caso, um mecanismo de sincronização também pode ser necessário se o receptor mudar de canal ao longo do tempo.

Para manter um determinado nível de QoS por longos períodos, pode ser necessário empregar estratégias dinâmicas de alocação de canal. Além da possibilidade de escolher os melhores canais, o mecanismo deve permitir uma rápida sincronização da rede em caso de mudança de canal. A alocação deve considerar a topologia da rede para evitar que sub-redes vizinhas usem os mesmos canais ou canais adjacentes, e algum mecanismo para estimar a qualidade dos canais, pois pode haver variações espaciais na qualidade.

2.1.2 Estimativa de qualidade de enlace

Os estimadores de qualidade de canal são a base para vários protocolos de roteamento, controle de topologia e mecanismos de alocação dinâmica de canais [Baccour et al. 2012]. Os estimadores devem ter boa reatividade às mudanças e manter uma boa estabilidade. Alguns deles são baseados em hardware usando o Indicador de Qualidade de Link (LQI) e Indicador de Intensidade de Sinal Recebido (RSSI), outros em software (PDR, número de retransmissões e assim por diante).

Todos os pacotes recebidos por rádios IEEE 802.15.4 têm um valor RSSI e LQI associado, entretanto, o RSSI também pode ser obtido independentemente da recepção do pacote. Nesse caso, o transceptor executa uma varredura de potência do canal, independentemente da fonte. Qualquer dispositivo que gere ruído e interferência influencia seu valor. O LQI, por outro lado, só pode ser medido durante a recepção dos pacotes, uma vez que sua métrica é baseada na análise dos primeiros símbolos dos pacotes recebidos. RSSI é mais estável que LQI, exceto em ambientes muito reflexivos, que mostram maior atenuação em pequena escala devido ao desbotamento em vários caminhos [Baccour et al. 2012].

Os estimadores baseados em software usam informações obtidas das camadas superiores, como PDR e o número necessário de transmissões de pacotes (RNP). O uso de métricas baseadas em PDR permite uma boa estimativa de enlace com qualidade muito alta ou muito baixa, mas apresenta alguns problemas em enlaces com qualidade intermediária. Quando a retransmissão é usada, as métricas baseadas em PDR podem superestimar a qualidade do enlace, pois não considera o número de tentativas de transmissão antes de uma recepção bem-sucedida. As métricas baseadas em RNP estimam o número necessário de transmissões de pacotes até a recepção bem-sucedida. ETX e Four-Bit (FB) são exemplos de estimadores baseados em RNP [Gomes et al. 2017].

Em [Gomes et al. 2017], os autores propuseram um LQE para IWSN, e um novo tipo de nó, o nó LQE, que estima a qualidade dos enlaces em tempo real usando RSSI e informações obtidas dos pacotes de dados recebidos. O LQE proposto é capaz de capturar os efeitos de multipercurso, interferência e assimetria dos enlaces.

Os experimentos foram realizados em ambiente industrial real utilizando rádios IEEE 802.15.4, e foram desenvolvidos modelos que permitem o uso de amostras RSSI para estimar a qualidade dos enlaces. Foi feita uma comparação com o Opt-FLQE, e os resultados mostraram que o estimador descrito é mais preciso e reativo para o tipo de ambiente em estudo. Ao contrário de outros LQEs na literatura, no LQE proposto, os nós

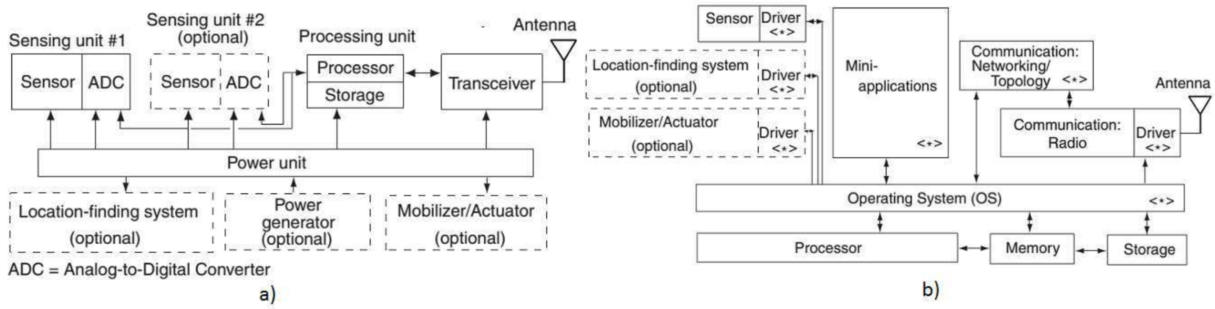

Figura 7: Hardware (a) e Componentes de Software (b), adaptado de [Sohraby, Minoli and Znati 2007].

sensores não precisam enviar pacotes de teste em difusão. Além disso, usando o nó LQE, os outros nós da rede não precisam interromper sua operação para monitorar a qualidade do enlace. Nessa solução, nenhum overhead é imposto aos nós finais e não há dependência do tamanho dos pacotes, como no LPED, por exemplo.

Este trabalho visa utilizar a ideia de LQE nos dispositivos de controle, como o nó sorvedouro e os CHs, para estimar a qualidade dos canais e inseri-los na lista de canais adequada. O LQE considera a assimetria dos enlaces, utiliza RSSI, o número de retransmissões e o PDR para estimar a qualidade dos canais.

2.2 Software e Hardware para WSN

Os nós sensores são geralmente compostos por uma ou mais unidades de detecção, uma fonte de alimentação operada com bateria ou conectada à empresa de distribuição de energia, uma unidade de processamento usada para processar, armazenar, criptografar e modular digitalmente, um transceptor e uma antena, conforme ilustrado em Figura 7 (a). No caso de dispositivos como o nó LQE, pode haver várias antenas para capturar informações de vários canais ao mesmo tempo.

Conectado ao hardware dos nós sensores, existem os seguintes subsistemas de software, conforme ilustrado na Figura 7 (b):

- Sistema operacional (SO): projetado para funcionar em dispositivos limitados em memória, potência, largura de banda e capacidade de processamento. Contiki, OpenWSN e TinyOS são três exemplos desse tipo de sistema operacional. Eles são de código aberto e conectam microcontroladores (MCU) pequenos, de baixo custo e baixa potência à Internet. A tabela 1 descreve os sistemas operacionais para dispositivos com recursos limitados;

Nome	Arquitetura	Escalonador	Modelo	MCU Suportada	Linguagem	Licença
TinyOS	Monolítico	Cooperativo	Dirigido a evento	AVR, MSP430	nesC	BSD
Contiki	Monolítico	Cooperativo/ Preemptivo	Dirigido a evento, Protothreads	AVR, MSP430, ARM	C	BSD
OpenWSN	Monolítico	Cooperativo	Event-driven	MSP430, ARM Cortex-M	C	BSD
RIOT	Monolítico/ Layered/ Microkernel	Preemptivo, tickless	Multi-threading	AVR, MSP430, ARM7, ARM Cortex-M, x86	C, C++	LGPLv2
FreeRTOS	Microkernel	Cooperativo/ Preemptivo	Multi-threading	AVR, MSP430, ARM, x86	C	GPL Modificada

Tabela 1: Sistemas Operacionais.

- Drivers: os drivers de sensor e transmissão são os módulos de software que gerenciam as funções básicas do sensor e os detalhes do enlace de transmissão do canal de rádio, incluindo o relógio e métodos de sincronização, codificação de sinal, recuperação de bits, contagem de bits, níveis de sinal e transceptores de modulação;

- Mini-aplicações: processamento de dados, armazenamento de valores de sinal e aplicações;

- Processadores de rede: gerenciam funções de transmissão, incluindo roteamento, buffering e encaminhamento de pacotes, manutenção de topologia, controle de acesso ao meio e criptografia..

Os processadores de sensores sem fio são comumente referidos como MCUs, e alguns dos recursos pelos quais esses MCUs são especialmente adequados para sistemas incorporados são sua flexibilidade em se conectar a outros dispositivos (como sensores), seu conjunto de instruções habilitado para capturar e processar sinais com requisitos de tempo críticos e seu consumo de energia normalmente baixo. Além desses recursos, o MCU tem a capacidade de reduzir o consumo de energia entrando em um estado de hibernação em que apenas partes do controlador estão ativas.

Os MCUs mais usados em protótipos de sensores sem fio incluem o processador da Atmel e da Texas Instruments. A tabela 2 mostra alguns exemplos e características do MCU de ambos os fabricantes.

É possível observar que algumas plataformas apresentam operação em banda dupla (2.4 GHz + 868/915 MHz), e a Tabela 3 mostra as informações sobre a camada física do padrão IEEE 802.15.4 e a frequência espectro. As bandas 868/915 MHz ISM poderiam

Plataforma	mica2	openmote-b	micaz	tmote sky	telosB
MCU	Atmel Atmega 128L	TI ARM Cortex-M3	Atmel Atmega 128L	TI MSP430	TI MSP430
Frequência	8 MHz	32 MHz	7.37 MHz	8 MHz	8 MHz
RAM (Kbytes)	4	32	4	10	10
Memória Flash	128	512	128	48	48
Radio	CC1000 315/433/ 868/916 MHz 38.4 Kbauds	CC2538 2.4 GHz 868/915 MHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4	CC2420 2.4 GHz 250 k *bit/s* 802.15.4
Operating Systems	TinyOS	OpenWSN Contiki	TinyOS	TinyOS Contiki	OpenWSN TinyOS Contiki

Tabela 2: Plataformas para RSSF.

Frequência spectrum	Modulação	Taxa de Bits	Taxa de Símbolos	Número de canais
868 MHz	BPSK	20 kbit/s	20 kbaud	1
915 MHz	BPSK	40 kbit/s	40 kbaud	10
2.4 GHz	O-QPSK	250 kbit/s	62.5 kbaud	16

Tabela 3: Camada física do IEEE 802.15.4

ser usadas como uma alternativa para mitigar os problemas com interferência, uma vez que são conhecidas por serem relativamente livres de interferência, mas várias tecnologias de rádio provaram causar problemas significativos para RSSF implantado na banda de 868 MHz [Barrenetxea et al. 2008] e frequências de 900 MHz [Kusy et al. 2011].

Embora a camada física escolhida dependa das regulamentações locais e da preferência do usuário, para os fins desta tese apenas a banda de 2,4 GHz não licenciada foi considerada.

2.2.1 Sistemas operacionais

Esta seção resume os Sistemas Operacionais (SO) mais conhecidos e utilizados em RSSF, com destaque para o Contiki e o OpenWSN, que suportam o dispositivo definido para os experimentos neste projeto.

TinyOS é um sistema operacional licenciado Berkeley Software Distribution (BSD) projetado para dispositivos sem fio de baixa potência e é um dos primeiros sistemas operacionais projetados especificamente para RSSF [Dargie and Poellabauer 2010].

O sistema operacional RIOT foi projetado para operações em tempo real, é baseado em um micro-kernel com suporte para multi-threading [Will, Schleiser and Schiller 2009]. Ele tem várias pilhas de rede, incluindo sua própria implementação da pilha 6LoWPAN (pilha GNRC), também o OpenWSN 6TiSCH[1], E a pilha de rede chamada Lightweight Content Centric Networking (CCN-lite) [Hahm et al. 2015].

O Contiki é um sistema operacional de código aberto projetado especificamente para RSSF. Seu kernel é orientado por eventos como o TinyOS, e o sistema inclui protothreads que fornecem uma abstração de programação semelhante a um thread, mas com pouca sobrecarga de memória. Protothreads no Contiki são uma combinação de alguns recursos de eventos e threads.

Outro exemplo de sistema operacional é o Berkeley OpenWSN [Watteyne et al. 2012]. É uma pilha de código aberto que visa implementar padrões sem fio de baixa potência, como IEEE 802.15.4e TSCH, 6LoWPAN, RPL e CoAP. Entre as plataformas que podem executar o OpenWSN estão GINA (AT86RF231), TelosB (CC2420), LPC (AT86RF231), K20 (AT86RF231) e OpenMote (CC2538).

2.2.2 Simuladores de rede

Existe um grande número de simuladores de rede que foram desenvolvidos para vários tipos de ambientes [Wehrle, Günes and Gross 2010]. Os cinco principais simuladores que suportam simulação de dispositivo de baixa potência, também chamados de Low-rate, Wireless Personal Area Network (LR-WPAN), são descritos nesta seção. Eles são amplamente utilizados na literatura devido aos recursos disponibilizados.

Os principais simuladores são TOSSIM, COOJA, OpenSim, NS2 e NS3 e OMNeT++. Em relação ao OMNeT++, este trabalho dá atenção especial a um de seus frameworks, Castalia[2], devido ao seu modelo realista baseado nos rádios transceptores CC2420 e CC1000, e ao modelo de canal wireless que desenvolvemos para simular o IWSN.

O ambiente de simulação Objective Modular Network Testbed (OMNeT++) é uma ferramenta de eventos discretos de código aberto baseada em C++ e usada para simulações de redes de comunicação, multiprocessadores e vários sistemas distribuídos. Consiste em módulos que se comunicam por meio de mensagens. Módulos simples podem ser agrupados para formar módulos compostos mais complexos. Um usuário define a estrutura de um módulo usando a linguagem de descrição de topologia OMNeT++ chamada NED.

O OMNeT++ é freqüentemente referido como um simulador de eventos discretos, mas na verdade, é um software que suporta vários modelos e estruturas de simulação

[1]IPv6 over the TSCH mode.
[2]https://github.com/boulis/Castalia. Accessed: 14/05/2020.

(frameworks), como o Framework INET, MiXiM e Castalia. Os frameworks são desenvolvidos independentemente do framework de simulação e seguem seus próprios ciclos de lançamento.

O Castalia é um simulador para RSSF, para detecção médica (Body Area Networks) e, geralmente, para redes de dispositivos embarcados de baixa potência. Pesquisadores e desenvolvedores podem usar este simulador para testar seus algoritmos e protocolos distribuídos em um canal sem fio realista com os modelos de rádio CC1000 e CC2420. Esse simulador usa o modelo de sombreamento log-normal como uma das formas de simular a perda média do percurso. Ele também modela a variação temporal da perda de caminho como um esforço para capturar fenômenos de desvanecimento em ambientes muito instáveis.

Uma descrição de um modelo simples e confiável para simular protocolos multicanais para IWSN utilizando o castalia é apresentado na seguinte seção e no artigo [Gomes et al. 2017], que captura os efeitos de desvanecimento, sombreamento e as características não estacionárias do canal. Esse modelo foi integrado ao simulador Castalia pelo autor deste trabalho, e também considera as diferenças nas características dos diferentes canais e a assimetria dos links.

2.2.3 Modelo de Simulação para IWSN

Na indústria, mudanças na topologia do ambiente, como a movimentação de uma grande estrutura metálica, podem causar mudanças nas características dos canais ao longo do tempo, levando a uma diferença no valor médio da potência recebida, mesmo com nós estáticos .

O canal sem fio em ambiente industrial pode permanecer por várias horas com as mesmas características, e após esse período pode ocorrer uma mudança brusca de canal [Agrawal et al. 2014]. Nesta seção, o modelo de canal desenvolvido para simular protocolos com múltiplos canais para IWSN é brevemente apresentado. Este modelo foi integrado ao simulador Castalia, que por padrão usa um único canal e não considera as características não estacionárias do canal por longos períodos.

Castalia é um simulador de eventos discretos desenvolvido em C++, baseado no *framework* de simulação de código aberto OMNeT++. Seu módulo de rádio é baseado em rádios reais para dispositivos embarcados de baixa potência, incluindo suporte para os transceptores CC2420 e CC1000, e o modelo de sombreamento log-normal, que fornece estimativas precisas para a perda média de percurso. O Castalia também é mais específico para redes com dispositivos com baixa capacidade de processamento do que outros simuladores, como o Network Simulator 3 (NS3) e OPNET, que se concentram em redes gerais.

Na camada física, o Castalia usa parâmetros reais de rádios IEEE 802.15.4 e um modelo baseado em sombreamento log-normal. O simulador também permite variação temporal na potência recebida usando amostras predefinidas que simulam o comportamento de um canal sujeito a desvanecimento por multipercurso. No entanto, por padrão o modelo implementado pelo simulador considera que o canal é estacionário, ou seja, as suas características permanecem inalteradas durante todo o tempo.

Para realizar a integração com o Castalia, foram feitas modificações na classe *WirelessChannel*, para capturar as variações nas características do canal ao longo do tempo. A duração média para alteraração dos parâmetros do canal para simular variações na qualidade do canal parece ser consistente com os resultados experimentais em [Agrawal et al. 2014] e [Wang and Yao 2013], em que o canal permanece estacionário por algumas horas. Os valores dos parâmetros para utilização dos modelos de sombreamento log-normal e Rice foram obtidos por meio de experimentos na indústria descritos em [Tanghe et al. 2008], considerando um cenário sem visada direta entre transmissor e receptor.

Para modelar o efeito do desvanecimento em um longo período, foi usada uma abordagem baseada na cadeia de Markov de dois estados. Um tempo médio de mudança (Tc) é definido para o modelo, em minutos, que é usado para definir a probabilidade de mudança p. Na implementação do modelo, o intervalo de mudança de estado da cadeia de Markov foi definido como igual a um minuto, o que resulta em $p = 1/T_c$. Com este parâmetro, é possível simular ambientes que permanecem inalterados por um longo período, bem como ambientes nos quais ocorrem mudanças mais frequentes na topologia. A distribuição de Rice foi usada para modelar a atenuação por múltiplos percursos, mas outras distribuições podem ser usadas, como a distribuição Nakagami-m. Diferentes modelos de atenuação podem ser facilmente integrados ao modelo de simulação.

Como os canais em ambientes industriais não estão correlacionados em frequência, para simular protocolos que usam canais múltiplos, uma instância do modelo pode ser usada para cada canal disponível para comunicação. No cenário estudado, a mudança de canais pode melhorar a qualidade de serviço da rede, uma vez que a rede é capaz de se adaptar às variações na qualidade do canal. Para permitir a avaliação de protocolos multicanais com Castalia, é necessário fazer modificações na interface entre o módulo que

Taxa de transmissão	1 pacote/s
Potência de transmissão	0 dBm
Tempo médio de mudança	30, 80 and 100 minutos
Expoente de perda de percurso (n)	1.69
Distância de referência (d_0)	15 metros
Perda de percurso na distância de referência ($L(d_0)$)	80.48 dB
Desvio padrão de sombreamento (X_σ)	8.13 dB
Fator Rice (K)	12.3 dB
Desvio padrão do fator Rice (K_σ)	5.4 dB

Tabela 4: Parâmetros usados nas simulações.

implementa o modelo de canal sem fio e os módulos que implementam os protocolos da camada superior.

Este modelo é descrito em detalhes em [Gomes et al. 2017], e para testá-lo valores aleatórios foram gerados usando uma biblioteca C++ chamada IT++[3] que realiza a simulação dos valores de potência recebidos em um receptor, com distância d = 20 m considerando os valores de n = 1,69, $d0$ = 15 m, $L(d0)$ = 80,48 dB e σ = 8,13 dB, que são aplicados no cálculo do desvanecimento em larga escala e o modelo de sombreamento log-normal. Esses valores foram obtidos a partir de experimentos em ambiente industrial descritos em [Tanghe et al. 2008].

A Tabela 4 apresenta os parâmetros utilizados para a simulação, que usa sombreamento lognormal e distribuição de Rice. Os parâmetros n, d_0, $L(d_0)$, e X_σ são usados para calcular a perda de percurso e sombreamento, e o parâmetro K é usado para calcular o desvanecimento em pequena escala com a distribuição de Rice. Para modelar as variações no nível de desvanecimento em pequena escala, também é possível configurar um valor para o desvio padrão do fator K (K_σ). Assim, cada vez que ocorre uma mudança nas características do canal, um novo valor de K é definido.

[3]IT++ Documentation - http://itpp.sourceforge.net - Acessado em 08/05/2020.

3 Padrão IEEE 802.15.4e

Vários estudos investigaram o desempenho do IEEE 802.15.4 e destacaram as limitações deste padrão, que não é adequado para aplicações que possuem latência e confiabilidade estritas em ambientes hostis como o industrial [Khanafer, Guennoun and Mouftah 2014], [Baronti et al. 2007], [Tennina et al. 2013], [Zhan, Xia and Anwar 2016], [Du, Navarro and Mieyeville 2015], e [Huang, Pang and Hung 2009].

Para superar as limitações do padrão IEEE 802.15.4, foram desenvolvidos e definidos cinco modos de operação IEEE 802.15.4e, também chamados de protocolos, a saber: TSCH, DSME, LLDN, AMCA e BLINK. Para os dois últimos, o padrão fornece apenas uma breve descrição, portanto, não é discutido neste capítulo.

Em geral, as melhorias no padrão IEEE 802.15.4e estão relacionadas ao suporte para comunicações multicanais, superframe mais flexível[1] (DSME) e o uso de um mecanismo de acesso livre de contenção baseado em TDMA, que reduz a colisão e minimiza o consumo de energia.

3.1 Time Slotted Channel Hopping (TSCH)

O modo TSCH foi desenvolvido para aplicações de automação de processos, usa EB, funciona em topologias em estrela, árvore e malha e emprega um mecanismo de salto de frequência. Ao contrário do DSME, que usa uma estrutura de multi-superframe periódico, o TSCH usa slotsframes periódicos, timeslots dedicados, e CSMA/CA com timeslots compartilhados em vez de acesso ao canal baseado em contenção (durante CAP). Além disso, o TSCH não implementa GACK.

No TSCH, cada nó pode obter informações de sincronização, sequência de salto de

[1]Período entre o envio do primeiro beacon, período de acesso de contenção (CAP), período de acesso livre de contenção (CFP) e o envio do segundo beacon.

frequência e timeslots e slotframes por meio do EB. O slotframe contém timeslots com base na contenção ou sem contenção. A atribuição de timeslots a dispositivos dentro do slotframe pode inicialmente ser comunicada por beacon, mas geralmente as camadas de rede superiores os configuram quando o dispositivo se junta à rede [Alderisi et al. 2015].

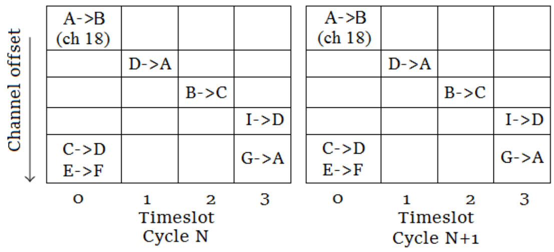

Figura 8: Examplo do TSCH slotframe.

A Figura 8 apresenta um exemplo de programação de enlace, em que há quatro timeslots, cinco desvios de canal e oito transmissões, graças à abordagem multicanal do TSCH. No exemplo, existem cinco links dedicados (A $\rightarrow$ B, D $\rightarrow$ A, B $\rightarrow$ C, I $\rightarrow$ D, e G $\rightarrow$ A) e um enlace compartilhado (C $\rightarrow$ D and E $\rightarrow$ F).

Por meio da sincronização de tempo e salto de frequência, o TSCH permite alta confiabilidade enquanto mantém os ciclos de execução (ciclos de trabalho) muito baixos e isso garante a eficiência energética. A sequência de salto é definida por um ID, o comprimento da sequência e uma lista ordenada de canais. Um enlace entre os dispositivos é identificado pelo intervalo de tempo e deslocamento do canal. Os enlaces compartilhados são atribuídos intencionalmente a vários dispositivos para transmissão. Isso pode levar a colisões e resultar em falha de transmissão detectada por um ACK ausente, mas um algoritmo de retransmissão de backoff (algoritmo CSMA/CA) pode ser implementado para enlaces compartilhados, para reduzir a probabilidade de colisões repetidas.

Cada timeslot na rede TSCH tem um deslocamento de canal associado, que é convertido em uma frequência em um método de salto pseudo-aleatório. Um enlace entre dois nós pode ser representado por um par entre timeslot no slotframe (n), e o deslocamento de canal usado pelos nós naquele intervalo de tempo, definido como [n,*channelOffset*]. A frequência f a ser usada para transmitir no timeslot n do slotframe pode ser calculada da seguinte forma:

$$f = FHS[(ASN + CO) \bmod FHS_L], \tag{3.1}$$

em que o número absoluto do slot (ASN) representa o número total de timeslots incrementados em cada timeslot desde o início da rede, % CO é a variável de deslocamento do canal usada pelos nós em um timeslot n, definido como [n, *channelOffset*], mod é o operador de módulo, FHS_L é o número de canais disponíveis, sem usar a llista de canais inadequados no cálculo. A partir daí a função FHS [.] gera um número entre 0 e 15, sendo o valor 0 representado pelo canal 11 (2,405 GHz) e 15 sendo representado pelo canal 26 (2,480 GHz), com diferença de 5 MHz entre os canais adjacentes.

Como no WirelessHART, no TSCH cada timeslot dura 10 ms, e permite a transmissão de um pacote de dados e do pacote de confirmação ACK correspondente. Se um determinado transmissor não receber o pacote ACK dentro do mesmo timeslot, ele pode retransmitir esse pacote no próximo timeslot alocado a ele, mesmo durante o mesmo slotframe, ou no próximo slotframe, dependendo da política de escalonamento atribuída a ele.

3.2 Lista de Canais Inadequados

É uma tarefa desafiadora configurar as listas de canais inadequados e, ao mesmo tempo, definir uma programação adequada das transmissões para que dois ou mais nós vizinhos não usem os mesmos canais ao mesmo tempo. Embora apresente uma melhoria no desempenho geral, o uso de listas de canais inadequados diminui a capacidade e a diversidade da rede, uma vez que o mesmo tráfego deve ser enviado por um menor número de canais.

Existem duas abordagens para a criação de listas de canais inadequados: global e local. No primeiro, todos os nós utilizam a mesma lista, mas essa solução tende a apresentar desempenho abaixo do ideal, pois os canais têm qualidade diferente nos diferentes enlaces da rede, mesmo que os nós sejam próximos uns dos outros [Gomes et al. 2017]. No método local, cada enlace usa diferentes conjuntos de canais [Kotsiou et al. 2017], e quanto mais dinâmico o ambiente, mais variada é a lista de canais inadequados. Apesar de ser melhor que a lista global em termos de desvanecimento seletivo de frequência, seu gerenciamento é muito mais complexo e está sujeito a problemas de interferência interna e colisões.

A transmissão paralela no mesmo canal resulta em vários nós transmitindo em difusão ao mesmo tempo, e se o ambiente for muito instável e a lista de canais inadequados for modificada com muita frequência, a lista local pode não ser útil devido ao excesso de quadros de sinalização necessários para a ressincronização .

Quando se trata de otimização de desempenho, a lista de canais, o roteamento e o escalonamento de tempo devem ser definidos em conjunto, porém, a união dos três métodos traz alta complexidade ao algoritmo [Gunatilaka, Sha and Lu 2017], dada a dificuldade em gerenciar lista de canais inadequados heterogêneos. A maioria das abordagens primeiro propõe a construção da lista e então modifica o escalonamento conforme necessário [Kotsiou et al. 2019].

Para criar uma lista de canais inadequados, que nos trabalhos publicados chamamos de denylist, é necessário analisar previamente os canais e, para isso, o uso de um LQE é indicado, que é um dispositivo dedicado ou uma função integrada no sensor. Para garantir um alto desempenho da rede, o processo de estimativa deve ser feito de forma contínua.

A importância do gerenciamento correto da denylist tem sido negligenciada pela maioria dos trabalhos quando se pesquisa o TSCH, que se concentram mais no escalonamento e no salto de frequência de forma cega, sem considerar a qualidade dos canais.

4 Adaptive-Blacklist TSCH

Para lidar com problemas de confiabilidade em IWSN, é necessário implementar mecanismos que permitam que a rede se adapte às variações que ocorrem ao longo do tempo na qualidade dos enlaces. Neste capítulo, é proposto um novo protocolo para IWSN, denominado Adaptive-Blacklist TSCH (AB-TSCH), que utiliza tanto salto de frequência quanto denylisting de canal, juntamente com a abordagem LQE para mitigar os problemas mencionados na tese.

4.1 Descrição do Protocolo AB-TSCH

AB-TSCH é um protocolo para RSSF voltado para ambientes industriais, embora possa ser usado em ambientes residenciais e de escritório. Baseia-se no método de salto de frequência estabelecido no protocolo TSCH IEEE 802.15.4e, e utiliza a ideia de denylist local semelhante ao padrão ISA110.11a, mas sem adiar a transmissão caso encontre um canal na denylist.

O protocolo AB-TSCH pode ser usado em redes com topologia em árvore ou estrela. A Figura 9 apresenta um exemplo de topologia em árvore estudada neste trabalho, que é mais complexa que a topologia em estrela devido à possibilidade de transmissões simultâneas, colisões e interferências internas.

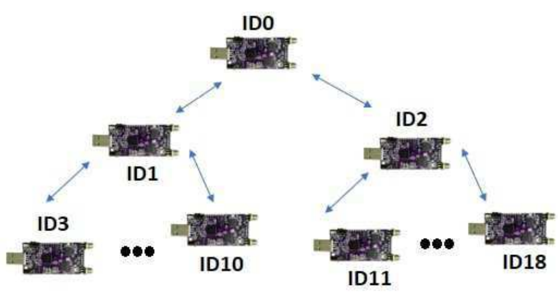

Figura 9: Nós identificados por seu ID, em que ID0 é o nó sorvedouro, ID1 e ID2 são CH na topologia de árvore, e os demais são nós finais.

Na configuração usada para avaliar o protocolo, 16 nós finais (ID3 a ID18) foram implantados e apenas detectam o ambiente e transmitem os dados. Na topologia estrela, o receptor é um nó sorvedouro que atua como coordenador da rede e LQE, e na topologia em árvore foram adicionados dois clusters (ID1 e ID2), que também funcionam como coordenadores e LQEs, cada um para sua própria sub-rede. Eles recebem os pacotes de dados dos nós finais e os encaminham para o nó sorvedouro (ID0).

Os nós finais transmitem apenas uma vez por timeslot e todos eles têm a mesma prioridade, com uma retransmissão no próximo slotframe, se necessário. Com o passar do tempo, os enlaces apresentarão denylists diferentes, e os canais permitidos não serão necessariamente os melhores, mas pelo menos não serão os piores canais em desempenho e confiabilidade.

No AB-TSCH, o acesso ao meio é baseado em uma estrutura chamada multislotframe, semelhante às estruturas definidas para os modos DSME e TSCH. Cada multislotframe é formado por um conjunto de slotsframes, e cada slotframe é composto por um conjunto de timeslots, em que o primeiro timeslot de cada slotframe é usado para a transmissão de EBs e os demais para pacotes de dados, conforme ilustrado na Figura 10.

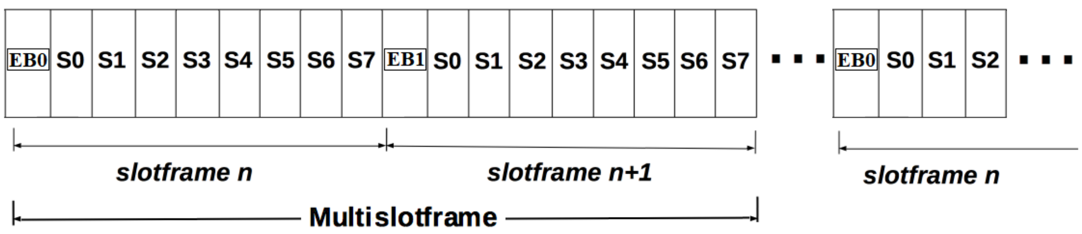

Figura 10: Multislotframe com vários timeslots, incluindo o timeslot EB no início de cada slotframe.

Após receber e processar o primeiro EB, representado como $EB0$, para sincronizar com a rede, os nós podem receber, mas não precisam processar os próximos EBs (EB1, EB2 e assim por diante) no mesmo multislotframe para manter a comunicação, a menos que eles percam a conexão. Com essa abordagem, o desempenho da rede é menos afetado por problemas de sobrecarga com o recebimento de EBs, em comparação com outros protocolos, como DSME e o modo habilitado para EB do padrão IEEE 802.15.4.

Todos os slotsframes dentro de um multislotframe têm a mesma estrutura, com os timeslots alocados para os mesmos nós, e a atribuição de timeslots é baseada nos requisitos da aplicação executada por cada nó final.

A transmissão dos EBs dos coordenadores da rede ocorre simultaneamente, em diferentes canais. Embora a lista de canais usados para transmiti-los seja a mesma para todos os coordenadores, cada um usa um deslocamento de canal diferente. O nó sorvedouro (ID0) transmite o primeiro EB usando o primeiro canal da lista (EB_T), com os CH ID1 e ID2 esperando para recebê-lo (EB_R). Após o recebimento, os CH encaminham o EB usando o segundo e o terceiro canais da lista (EB_T), respectivamente, no próximo timeslot. Este processo é ilustrado na Figura 11. Antes que as redes sejam iniciadas, os CH precisam receber o EB do nó sorvedouro e, em seguida, encaminhá-lo para os nós finais.

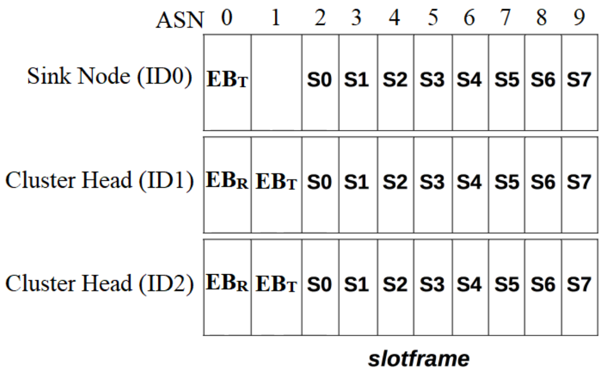

Figura 11: Slotframe com EBs na topologia em árvore.

Todos os pacotes de dados e pacotes ACK transmitidos têm um campo de canais denylisted que o transmissor e o receptor estão cientes de não usar. Nesse caso, EB são usados apenas para sincronização, não para transmitir a denylist. Isso pode ser justificado pelo fato de que, ao perder a sincronização e, em seguida, ressincronizar usando os seguintes EBs, dependendo do tamanho da rede, as informações denylist inseridas no EB podem estar desatualizadas em redes muito dinâmicas. Para sincronizar com a rede, o nó salta por todos os canais, aguardando o EB.

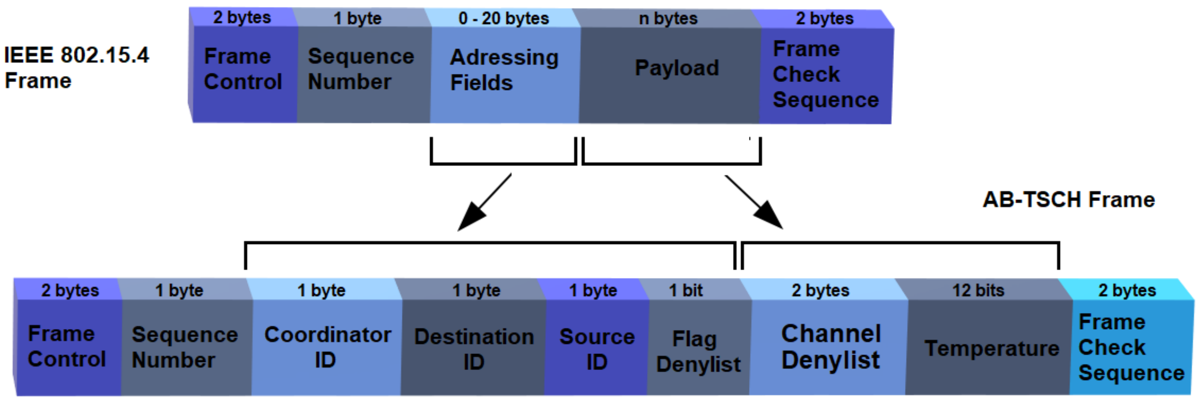

Figura 12: Pacote de dados do padrão IEEE 802.15.4 e e do AB-TSCH.

A Figura 12 ilustra a estrutura do pacote de dados transmitido pelos nós finais. Os campos de endereçamento são usados para armazenar o ID do coordenador, o ID do destino, o ID da fonte e o sinalizador de denylist, que é usado para informar ao receptor que o transmissor está ciente sobre a denylist atualizada.

A carga útil consiste em dois campos, o primeiro representando o denylist com 16 canais (16 bits), e o segundo com 12 bits representando a temperatura, em que 11 bits são definidos para os dados e 1 bit para o operador (+ ou -) Por exemplo: com canais 3, 9 e 15 na denylist e a temperatura monitorada em cerca de -15° C, a carga útil é a mesma que $1000\ 0010\ 0000\ 1000_2$ para o canal denylist, $0111\ 1101\ 1111\ 0111_2$ para a lista de canais adequados (canais 0, 1, 2, 5, 6, 7, 8, 10, 11, 12, 13, 14 e 15), o operador 1_2 devido ao valor negativo da temperatura e $000\ 0000\ 1111_2$ para a temperatura.

Após a primeira análise do LQE, um canal é listado como inadequado e o receptor insere no pacote ACK as informações da denylist do canal após ter recebido o pacote de dados do transmissor. Uma vez que o transmissor recebeu o pacote ACK, ele atualiza seu próprio campo denylist com aquele canal, e transmite o novo pacote de dados com as informações atualizadas e a flag 1_2 ativada. Depois que o destino recebeu esse pacote, ele o compara com o pacote ACK mais recente para saber se o transmissor está ciente do canal na denylist. Se os resultados forem confirmados, o receptor envia o pacote ACK para informar ao transmissor que agora ele pode se comunicar usando o novo canal de denylist. Este procedimento é ilustrado na Figura 13.

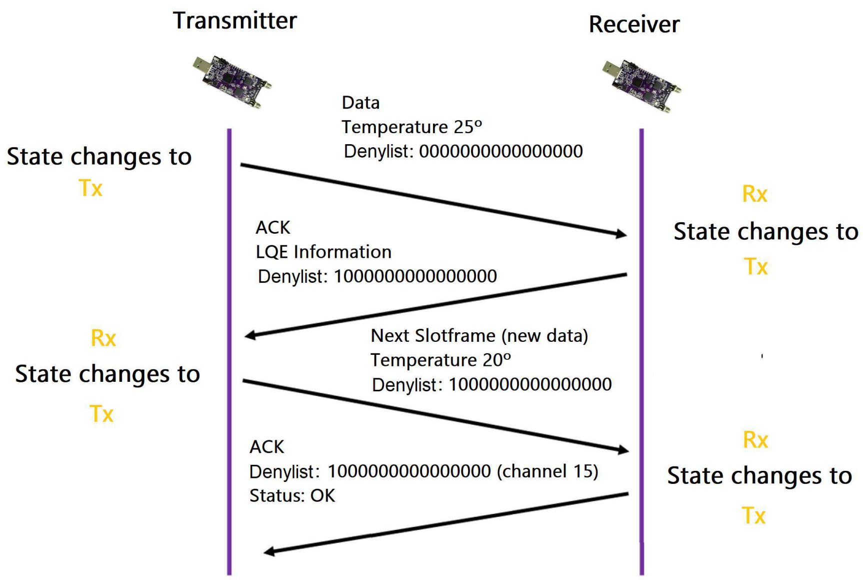

Figura 13: Procedimento de transmissão da informação denylist dos nós LQE para os nós finais.

A abordagem de deslocamento com múltiplos canais é usada para aumentar as chances de gerar um canal com boa qualidade, bem como para evitar a sobreposição de frequência

```
int offset1[8] = {0,1,4,5,8,9,12,13};
int offset2[8] = {2,3,6,7,10,11,14,15};
```

Figura 14: Deslocamentos de canal de cada CH, para os nós escolherem canais espectralmente distantes.

com vizinhos que transmitem no mesmo timeslot. No entanto, há uma grande probabilidade de nenhum dos deslocamentos atribuídos gerar um canal na lista de permissões, especialmente com denylists grandes. Como o tamanho da denylist depende de fatores externos, o ponto-chave na denylist local é quantos deslocamentos de canal devem estar disponíveis.

Neste protocolo, uma abordagem híbrida é definida para atribuir deslocamentos de canal. Um único deslocamento é definido para as comunicações entre o nó sorvedouro e os CHs, uma vez que não transmitem ao mesmo tempo (cada um tem seu próprio timeslot), e oito offsets de canal definidos estaticamente para cada enlace entre os CHs e os nós finais, de maneira que duas transmissões paralelas usam canais não adjacentes diferentes, conforme ilustrado na Figura 14.

4.2 Resultados das Simulações

Os experimentos foram realizados utilizando topologias em estrela e em árvore. Na topologia em estrela, quanto maior a lista de canais bloqueados (denylist - DL), maior o desempenho da rede, pois só são mantidos os melhores canais. Na topologia em árvore, existe um limite para a quantidade de canais bloqueados, pois há dispositivos com transmissões simultâneas e o desempenho pode ser reduzido caso um número de canais bloqueados seja atribuído.

No exemplo ilustrado, a rede em estrela com nove canais bloqueados apresentou desempenho superior. Mais do que esse número, a rede passa a desempenhar de forma negativa.

Além desse estudo, também foi comparado o desempenho entre a versão legada do TSCH, e a solução proposta pelo AB-TSCH, que apresentou desempenho superior, conforme ilustrado na Figura 16.

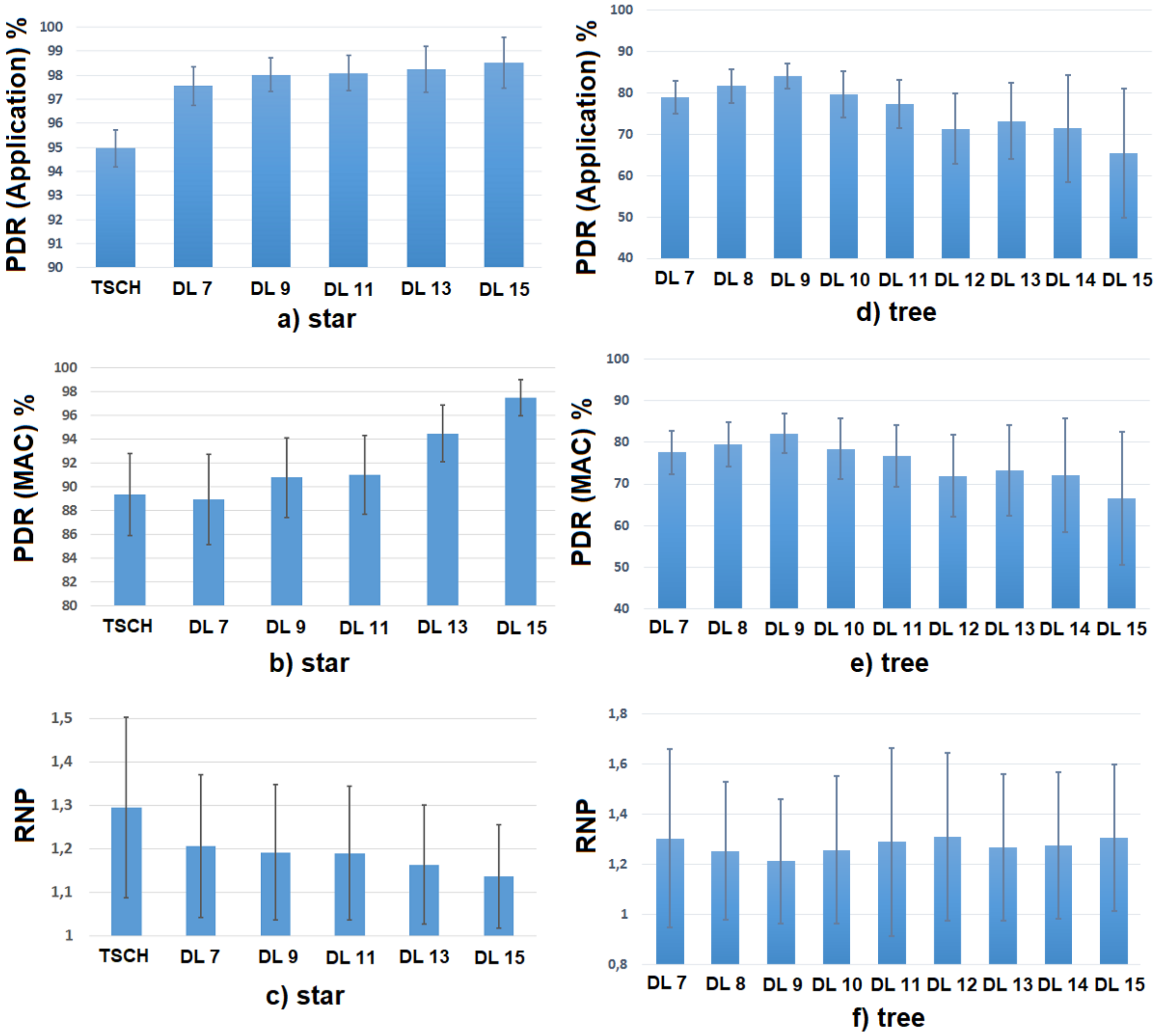

Figura 15: Resultados da topologia em estrela: a) *PDR na camada de aplicação*, b) *PDR na camada MAC* and c) *RNP*. Resultados na topologia de árvore: d) *PDR na camada de aplicação*, e) *PDR na camada MAC* and f)*RNP*.

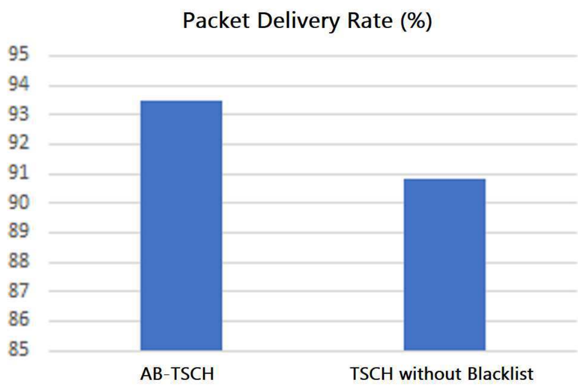

Figura 16: PDR médio de AB-TSCH com denylist de canal e sem colisão, e o TSCH sem denylist e sem colisão.

5 Lista tripla de canais usando lógica difusa

Neste capítulo, é proposto o uso de três listas de canais para gerenciar a atribuição de canais: uma denylist para indicar canais que não devem ser usados, uma lista de canais adequados chamada allowlist e uma greylist de canais com qualidade incerta. É proposto um método de lógica fuzzy para classificar os canais e incluí-los na lista apropriada, bem como um mecanismo de gerenciamento de deslocamento de canal adequado para evitar colisões.

Conforme mencionado, ao se referir à lista de canais, o conceito mais utilizado é o de lista negra e, ao contrário das listas negras, existem as listas brancas. No entanto, o conceito de lista negra ou denylist possui uma lacuna; em uma whitelist, ou allowlist, não há dúvida de que um canal é bom, mas em uma denylist pode haver alguma dúvida se esse canal é realmente ruim porque, se um estimador de qualidade defeituoso for usado, é possível que um canal com qualidade intermediária seja bloqueado.

Muitos conceitos do mundo real não podem ser bem representados usando limites claramente definidos. L. A. Zadeh desenvolveu a teoria dos conjuntos fuzzy, que generaliza a teoria dos conjuntos clássicos para permitir que os objetos tenham graus de pertinência a certos conjuntos, permitindo a representação de conceitos vagos e imprecisos, mantendo a precisão matemática no tratamento [Zadeh 1965].

A maioria dos problemas encontrados na vida real pode ser resolvida com base em informações imprecisas, incompletas e vagas que estão disponíveis no momento, e ainda assim as pessoas são capazes de resolver esses problemas de forma satisfatória. Isso se deve ao processo de raciocínio aproximado a partir dessas informações, para se obter um resultado aproximado, mas satisfatório para aquele problema.

Expresso em termos de uma instrução*if-then*, em que a parte 'if' é o antecedente e a parte 'then' é o consequente, o problema tenta avaliar se um canal deve ser usado ou não, dependendo de as métricas fornecidas para estimar a qualidade do enlace.

No domínio de RSSF, há uma grande imprecisão para definir quando uma rede é de boa ou má qualidade. Mais precisamente, pode haver tendências representadas por curvas, e uma dessas curvas pode ser definida usando uma distribuição gaussiana, por exemplo.

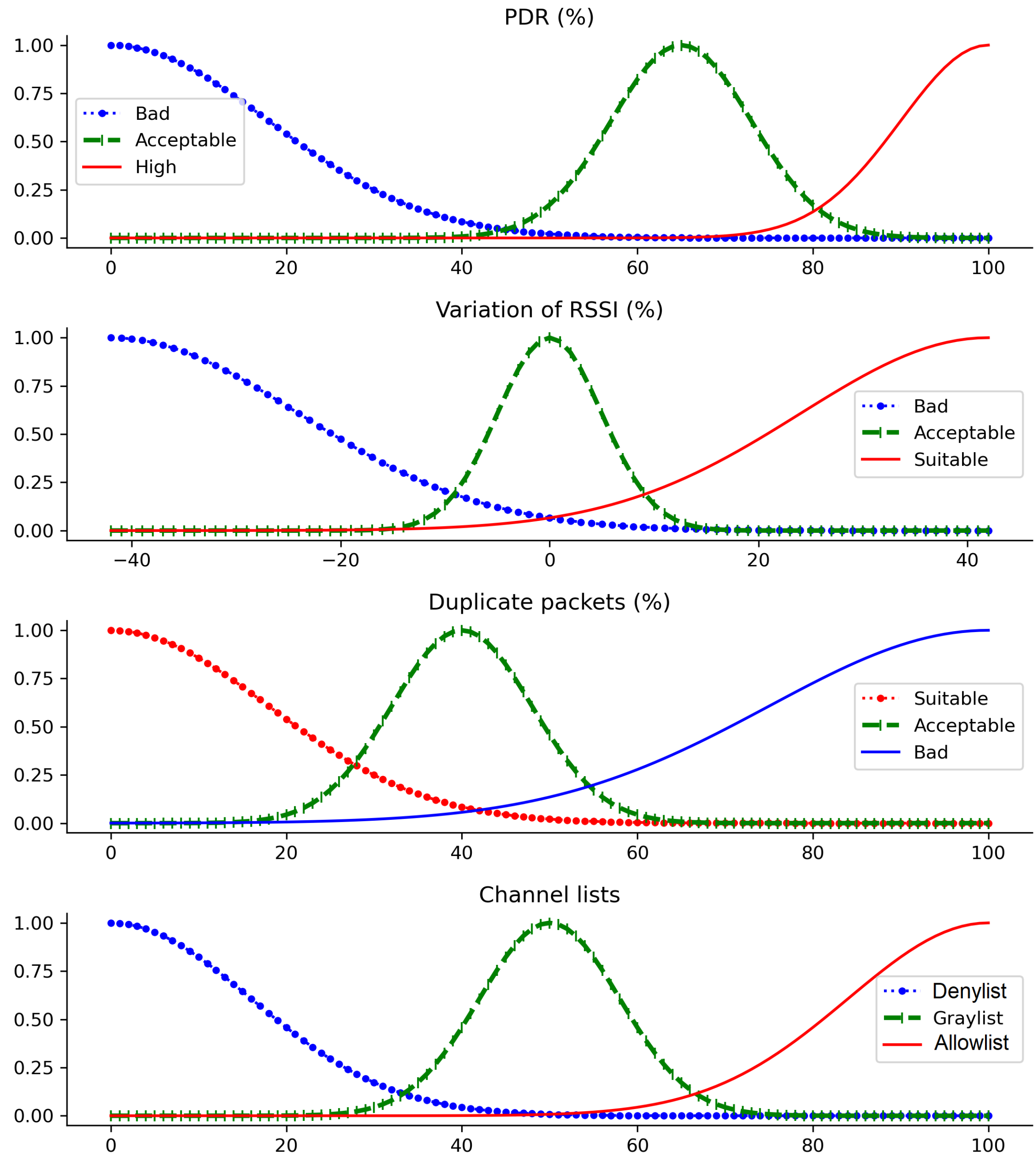

Figura 17: Sistema de controle difuso em RSSF com PDR, a variação do RSSI e o número de pacotes duplicados como entrada e as listas de canais como saída.

No exemplo deste trabalho, conforme ilustrado na Figura 17, três métricas são usadas para definir a lista de canais onde um determinado canal será temporariamente inserido. O sistema de controle da proposta estabelece que quanto maior o PDR, maior o percentual de sucesso e o canal tem alta qualidade. Quando o PDR atinge cerca de 80%, fica a dúvida sobre a qualidade do canal se ela está alta ou aceitável.

O mesmo ocorre nas outras métricas. A variação do valor médio de RSSI em relação ao melhor valor médio de RSSI do ciclo anterior é a segunda métrica. O valor RSSI médio se refere ao momento em que o LQE recebe 100 pacotes. Se no ciclo seguinte o valor médio for reduzido, indica que houve uma piora na qualidade do enlace em relação ao período anterior. Se essa variação estiver acima do valor estabelecido para a rede, pelo método da lógica fuzzy, é possível que o canal tenha mudado de perfil e agora seja considerado aceitável ou mesmo de má qualidade.

A última métrica está relacionada ao número de pacotes duplicados. Quando um nó final envia um pacote ao CH, mas não recebe um ACK dele para confirmar que foi recebido, o nó final entende que o pacote provavelmente não foi recebido pelo CH e reenvia o mesmo pacote em o próximo intervalo de tempo. Em alguns casos, o CH realmente recebeu o pacote, enviou de volta o pacote ACK para o nó final, mas não o recebeu. Ao receber o mesmo pacote de dados no próximo timeslot, o CH entende que já foi enviado e o considera duplicado. Isso pode indicar um problema com a conexão downstream do enlace, e para que a rede tenha alto desempenho, ambos os lados da comunicação precisam apresentar boa qualidade, não apenas a direção upstream. No exemplo da Figura 17, se cerca de 27% dos pacotes transmitidos forem duplicados, fica a dúvida se o perfil do canal é adequado ou aceitável, e em torno de 70% não há dúvida que o canal está apresentando má qualidade em relação a pacotes duplicados. A decisão sobre em qual lista o canal será inserido está ilustrada no último exemplo da Figura 17.

Para modelar este problema, existe uma entrada para o sistema, também chamada de antecedente, que é formada a partir do PDR, a variação do RSSI, e o número de pacotes duplicados, os consequentes, ou seja, a saída com as listas de canais, e as regras de decisão são definidas da seguinte forma:

- **Antecedent (input)**
  1. **PDR**
     - Universe: Scale of 0% to 100%;
     - Fuzzy set: bad, acceptable, high;
  2. **RSSI variation**
     - Universe: Scale of -42% to 42%;

  - Fuzzy set: bad, acceptable, suitable;

3. **Duplicate packets**
   - Universe: Scale of 0% to 100%;
   - Fuzzy set: suitable, acceptable, bad;

- **Consequent (output)**
  - **Channel list**
    * Universe: Scale of 0% to 100%;
    * Fuzzy set: denylist, greylist, allowlist;

O universo da parte consequente está relacionado com a qualidade geral da rede e suas respectivas listas de canais. Quanto mais próximo de 100%, de acordo com as regras de decisão, maior a probabilidade de um determinado canal ser inserido na lista de de canais adequados, e quanto menor essa porcentagem, maior a probabilidade de ser inserido na lista de canais inadequados. A questão é se o canal não é bom nem ruim o suficiente, então é provável que seja inserido na lista cinza.

- **Decision Rules**
  - IF the *PDR* was high and *RSSI variation* was (suitable or acceptable) THEN the *Channel list* will be allowlist
  - IF the *PDR* was bad or *RSSI variation* was bad THEN the *Channel list* will be denylist
  - IF the *PDR* was bad and *RSSI variation* was acceptable THEN the *Channel list* will be denylist
  - IF the *PDR* was (high or acceptable) and *RSSI variation* was (bad or acceptable) THEN the *Channel list* will be greylist
  - IF the *PDR* was acceptable and *RSSI variation* was (acceptable or great) THEN the *Channel list* will be allowlist
  - IF the *PDR* was (acceptable or high or bad) and *Duplicate packets* was bad THEN the *Channel list* will be denylist
  - IF the *Duplicate packets* was acceptable and *RSSI variation* was (acceptable or suitable) THEN the *Channel list* will be greylist

– IF the *Duplicate packets* was suitable and *RSSI variation* was bad THEN the *Channel list* will be allowlist

Além da topologia em estrela e topologia em árvore com dois CH estudadas no Capítulo 4, uma topologia maior foi estudada com quatro CH e 48 nós finais, em que cada CH está diretamente conectado a 12 nós finais diferentes , conforme ilustrado na Figura 18. Isso significa que na topologia com dois CH, dois enlaces (4 nós finais) têm transmissões paralelas em cada timeslot, e na topologia com quatro CH, quatro enlaces (8 nós finais) têm transmissões paralelas.

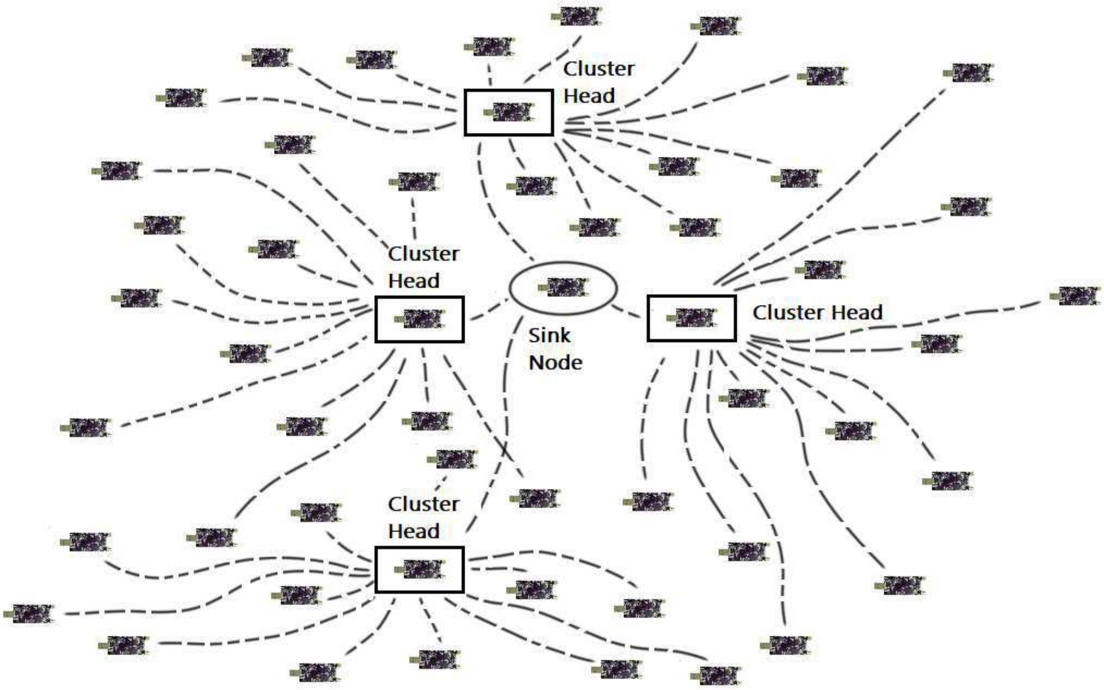

Figura 18: Uma das topologias com quatro CH.

Em relação aos campos dos pacotes, todo pacote de dados e ACK transmitido possui um campo de canais denylisted e greylisted que o transmissor e o receptor não têm permissão de utilizar (denylist) ou podem usar apenas em transmissões posteriores (greylist).

A Figura 19 ilustra a estrutura do pacote de dados transmitido pelos nós finais, e é muito semelhante à estrutura do AB-TSCH, e inclui o campo da lista cinza de canais com 2 bytes. As listas de canais são inseridas nos pacotes ACK transmitidos pelos receptores (LQEs) e nos pacotes de dados transmitidos pelos nós. Inicialmente, o pacote de dados possui apenas canais permitidos, mantendo os bits zerados nos campos deny e greylist, até que a análise seja feita pelo LQE e as listas sejam informadas no pacote ACK e estabelecidas no próximo pacote de dados.

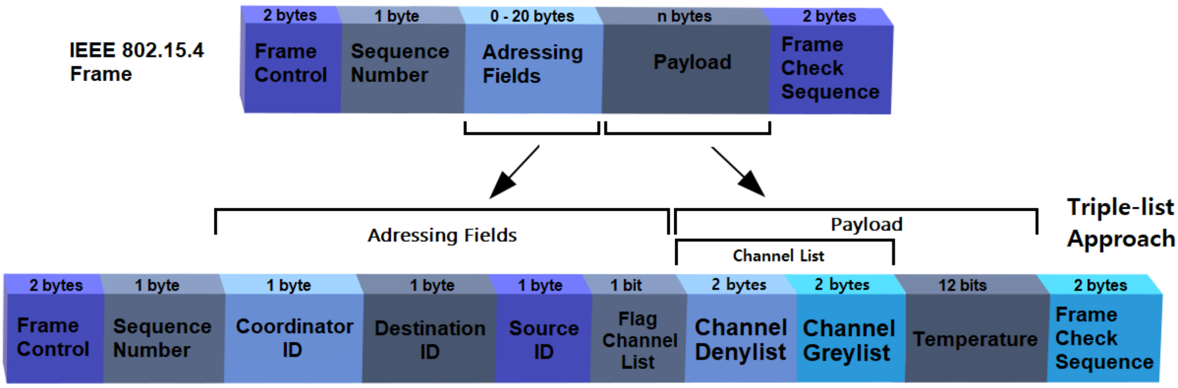

Figura 19: Pacote de dados do padrão IEEE 802.15.4 e a proposta deste trabalho.

5.1 Resultados

Os nós foram dispostos em uma área de 200 x 200 m, e três repetições foram executadas para as configurações avaliadas. Em cada replicação, a rede operou por duas horas, e a posição dos nós foi atribuída aleatoriamente. Para garantir um resultado justo, para cada repetição foi utilizada a mesma semente (a semente é um parâmetro do simulador) para avaliar os protocolos, sendo utilizadas sementes diferentes nas diferentes repetições. Assim, os protocolos foram avaliados considerando os nós na mesma posição e com as mesmas características para o canal sem fio durante as replicações.

A Figura 20 com os rótulos explicados na Tabela 5, mostra o desempenho em termos de PDR na aplicação e nas camadas MAC de ambas as configurações. Os experimentos avaliaram o TSCH em um canal de salto cego, com desvios de canal espectralmente distantes uns dos outros para evitar que sensores com transmissões paralelas enviem dados em canais semelhantes ou adjacentes. Esta é a configuração mais usada na literatura para o protocolo TSCH, que não usa denylists nem usa estimadores de qualidade de canal.

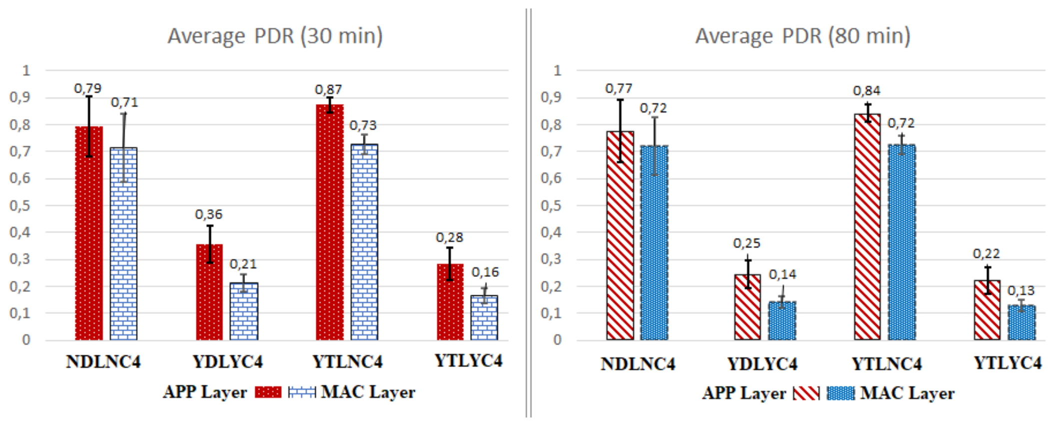

Figura 20: PDR nas camadas de aplicação e MAC em dois períodos diferentes de mudança do parâmetro da cadeia de Markov.

Label	Meaning
NDLNC4	Experimento sem denylists e sem colisões com 4 cluster heads
YDLYC4	Experimento com denylists e colisões com 4 cluster heads
YTLNC4	Experimento com lista tripla e sem colisões com 4 cluster heads
YTLYC4	Experimento com lista tripla e colisões com 4 cluster heads

Tabela 5: Explica os rótulos usados nos experimentos

A diferença de desempenho é perceptível quando o problema de colisões é levado em consideração, especialmente em grandes redes com transmissões simultâneas. O desvio padrão também é alto para redes que não usam listas de canais, seja por lista dupla ou lista tripla. Nas três abordagens que usaram listas de canais, a variação foi pequena em comparação com a abordagem cega.

Nos experimentos sem o gerenciamento de deslocamento de canal, o PDR na camada de aplicação foi muito baixo, de 22% (80 minutos) a 36% (30 minutos). Sem colisões, o PDR ficou entre 77% (80 minutos) e 87% (30 minutos), conforme mostrado na Figura 20.

6 Conclusions and Proposals for the Future

Este capítulo está organizado da seguinte forma: a Seção 6.1 apresenta o resumo desta tese, e a Seção 6.3 apresenta orientações de pesquisas futuras relacionadas a este tópico.

6.1 Resumo desta Tese

Nesta tese, duas implementações foram propostas para IWSN para mitigar os efeitos negativos de ambientes industriais no meio sem fio. Duas topologias principais foram estudadas, com os métodos desenvolvidos, e avaliadas por meio de simulações, utilizando um modelo de canal realista que simula o comportamento de um ambiente industrial.

Na aplicação desses métodos, foram considerados problemas típicos do ambiente industrial, como sombreamento, atenuação, variações espaciais na qualidade do canal e o comportamento não estacionário do canal sem fio em longos períodos. Foi discutido o uso de protocolos multicanais e alocação dinâmica de canais pela atribuição de listas de canais para melhorar a qualidade do serviço de RSSF. A abordagem multicanal apresenta várias vantagens se comparada à abordagem de canal único.

Além disso, foi demonstrada a necessidade de usar estimadores de qualidade, que analisam os canais antes que os dispositivos sejam autorizados a utilizá-los, evitando assim perdas de pacotes e desperdício de energia.

Dada a complexidade de adicionar outro elemento à rede, seja por hardware com dispositivo dedicado, seja por software, com função embutida nos nós coordenadores, a maioria dos estudos não utilizou esse tipo de abordagem. Porém, a vantagem é que o LQE é inserido apenas nos nós coordenadores, deixando os nós finais apenas para monitorar o ambiente e transmitir os dados capturados para o nó sorvedouro.

Além do PDR, informações sobre pacotes duplicados na rede e a variação do RSSI também são utilizadas para estimar a qualidade do canal, inclusive na direção reversa do enlace, para identificar problemas de assimetria. Com esta solução, nenhum overhead é imposto aos nós finais da rede e não há aumento no tráfego, ao contrário de outras soluções encontradas na literatura, que utilizam pacotes de beacon ou incluem redundância nos pacotes.

Uma abordagem também foi usada para diminuir a latência das comunicações e aumentar o throughput da rede. Essa abordagem causa algumas perdas de pacote, mas diminui o atraso da rede, permitindo que os dispositivos enviem mais dados ao sorvedouro e, assim, compensando pequenas perdas com essa abordagem.

O uso de listas de canais mostrou-se positivo em relação à abordagem cega, que não possui estimadores. Tanto nas listas duplas quanto nas triplas, o desempenho foi superior às configurações avaliadas. Uma característica importante que diferencia as topologias em estrela e em árvore é que nesta última foram definidas as transmissões simultâneas, aumentando a complexidade, sendo necessário levar em consideração os canais pelos quais os dados são transmitidos. Os canais semelhantes e adjacentes precisam ser evitados para reduzir as colisões e interferências internas.

Na rede estrela, não foram definidas transmissões paralelas, portanto, quanto maior a listas de canais inadequados, maior a qualidade da rede, mais pacotes são entregues, pois esta abordagem permite que cada dispositivo use apenas o melhor canal. Porém, em uma rede em árvore nem sempre é possível garantir que os dispositivos utilizem apenas o melhor canal, pois o melhor canal de um enlace também pode ser o melhor canal de outro enlace que transmite em paralelo, o que gera colisões e acaba superestimando ou subestimando a qualidade real do canal.

Por este motivo, foram atribuídos diferentes desvios de canal, para que os nós sempre utilizem canais diferentes no mesmo timeslot. O protocolo AB-TSCH leva esse problema em consideração e usa salto de frequência com diferentes deslocamentos de canal, além de receber ajuda do LQE para identificar quais canais são mais adequados para uso temporário.

Além do AB-TSCH que usa uma lista de canal duplo, e dada a imprecisão sobre a qualidade real de um canal, uma abordagem de lista tripla de canais foi proposta. Ela permite que canais de qualidade duvidosa sejam utilizados apenas em casos especiais, como quando acontecem grandes perdas de pacotes da rede em geral, por exemplo. Os melhores canais são sempre usados até que tenham qualidade reduzida e sejam bloqueados ou inseridos na lista cinza.

Para evitar que vários canais sejam bloqueados, reduzindo assim a diversidade, um número máximo de quatro canais foi estabelecido para a lista de canais inadequados. Quando houver mais canais com baixa qualidade, os mais antigos inseridos na lista inadequada são removidos e inseridos na greyist, e os mais novos inseridos na lista inadequada, da mesma forma que uma abordagem FIFO (First In First Out).

Nesta tese, duas e três listas de canais foram estudadas. A abordagem com três listas de canais apresentou desempenho superior, porém ela aumenta a complexidade da rede e requer um campo adicional no quadro de dados. É possível que listas adicionais aumentem a precisão do modelo, mas é importante ter em mente que a complexidade e a sobrecarga também aumentam, o que pode comprometer o rendimento e o consumo de energia.

6.2 Lista de publicações

6.2.1 Artigos de Conferência

- Diego V. Queiroz, Ruan D. Gomes, Cesar Benavente-Peces. Performance Evaluation of Default Active Message Layer (AM) and TKN15.4 Protocol Stack in TinyOS 2.1.2. In Proceedings of the 6th International Conference on Sensor Networks - Volume 1: SENSORNETS, 69-79, February/2017, Porto, Portugal. ISBN: 978-989-758-211-0, DOI: 10.5220/0006204200690079

- Ruan Delgado Gomes, Marcelo Alencar, Diego Véras de Queiroz, Iguatemi Eduardo da Fonseca, César Benavente-Peces. Comparison between Channel Hopping and Channel Adaptation for Industrial Wireless Sensor Networks. In Proceedings of the International Conference on Sensor Networks, At: Porto, Portugal, February/2017. DOI: 10.5220/0006206800870098

- Marcelo Alencar, César Benavente-Peces, Iguatemi Fonseca, Diego Véras de Queiroz, Ruan Delgado Gomes. Avaliação do Uso de Adaptação de Canal e Estimação de Qualidade de Enlace em Redes TSCH, XXXVI Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, December 2017, DOI: 10.14209/sbrt.2018.190.

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Evaluation of Channels Blacklists in TSCH Networks with Star and Tree Topologies. Q2SWinet'18: Proceedings of the 14th ACM International Symposium on QoS and Security for Wireless and Mobile Networks, October/2018, Pages 116–123, DOI: 10.1145/3267129.3267131.

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Join Multiple Channels and IEEE 802.15.4e TSCH Protocol Use Effects on WSN Performance and Energy Efficiency, 9th International Conference on Pervasive and Embedded Computing and Communication Systems, December 2018, DOI: 10.5220/0008162300830090.

- Gomes, R. D., Queiroz, D. V., Fonseca, I. E., Alencar, M. S., Benavente-Peces, C. Short course: Redes de Sensores sem Fio Industriais. In: XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2017, São Pedro, SP (short course proposal accepted).

6.2.2 Artigos de Revistas

- Diego Véras de Queiroz, Marcelo Sampaio de Alencar, Ruan Delgado Gomes, Iguatemi Eduardo da Fonseca, César Benavente-Peces. Survey and Systematic Mapping of Industrial Wireless Sensor Networks, August 2017. Journal of Network and Computer Applications. DOI: 10.1016/j.jnca.2017.08.019. Qualis 2016 ENGENHARIAS IV: A1

- Queiroz D.V., Benavente-Peces C., Gomes R.D. (2019) TinyOS-Based WSN Performance: Default Active Message Layer vs. TKN15.4. In: Benavente-Peces C., Cam-Winget N., Fleury E., Ahrens A. (eds) Sensor Networks. Communications in Computer and Information Science, vol 1074. Springer, Cham. DOI: 10.1007/978-3-030-30110-1_1. Qualis 2016 ENGENHARIAS IV: B5

- Gomes, R., Queiroz, D., Fonseca, I., & Alencar, M. (21-05-2017). A Simulation Model for Industrial Multi-Channel Wireless Sensor Networks. Journal of Communication and Information Systems, 32(1). DOI: 10.14209/jcis.2017.4. Qualis 2016 ENGENHARIAS IV: B1

- Ruan Delgado Gomes, Diego Véras de Queiroz, Abel Cavalcante Lima Filho, Marcelo Sampaio de Alencar. Real-Time Link Quality Estimation for Industrial Wireless Sensor Networks Using Dedicated Nodes. Ad Hoc Networks, Volume 59, 1 May 2017, Pages 116-133, Ad Hoc Networks. DOI: 10.1016/j.adhoc.2017.02.007. Qualis 2016 ENGENHARIAS IV: A1

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. Channel Assignment in

TSCH-based Wireless Sensor Networks using Fuzzy Logic. **Submitted to Journal of Ambient Intelligence and Humanized Computing (2020). Qualis 2016 CIÊNCIA DA COMPUTAÇÃO: B1**

6.2.3 Capítulo de Livro

- Diego Véras de Queiroz, Ruan Delgado Gomes, César Benavente-Peces, Iguatemi Eduardo da Fonseca, Marcelo Sampaio de Alencar. In book: Nanosensors for Smart Cities - Micro and Nano Technologies, 2020, Pages 515-526. DOI: 10.1016/B978-0-12-819870-4.00029-3

6.3 Direcionamentos Futuros de Pesquisa

As duas soluções propostas apresentaram desempenho superior em termos de entrega e determinismo de pacotes e, em trabalhos futuros, as soluções serão avaliadas em resultados experimentais com dispositivos reais em diferentes tipos de ambientes industriais e para diferentes tipos de aplicações, com diferentes prioridades de pacotes.

Como a rede em topologia em árvore adiciona mais complexidade à solução, não só pelas transmissões simultâneas, mas também pela escalabilidade, uma vez que a rede em árvore permite uma expansão maior da rede do que a topologia em estrela, esta última topologia não será avaliada nos experimentos.

A topologia em malha, também proposta pelo padrão TSCH, adiciona ainda mais complexidade, pois permite maior expansão da rede do que a topologia em árvore, e também permite rotas redundantes (embora a rede em árvore permita, mas é menos comum que a rede em malha). Rotas redundantes usando listas de canais, com diferentes deslocamentos de canal tornam a tarefa ainda mais desafiadora, serão avaliadas em trabalhos futuros.

O uso de estimadores mais precisos também será avaliado, envolvendo não só a análise de RSSI, a variação de RSSI e pacotes duplicados. Esses estimadores serão classificados segundo a abordagem proposta na tese, por meio da lógica fuzzy.

Outro trabalho futuro será o desenvolvimento de algoritmos para planejamento automático de redes, que definem qual topologia usar e quantos timeslots atribuir para cada nó sensor. O uso de múltiplos transceptores no nó sorvedouro também será avaliado para reduzir a latência, permitindo assim que os cluster heads transfiram pacotes para o nó sorvedouro simultaneamente.

Referências

[Agrawal et al. 2014]AGRAWAL, P. et al. Long Term Channel Characterization for Energy Efficient Transmission in Industrial Environments. *IEEE Transactions on Communications*, v. 62, n. 8, p. 3004–3014, aug 2014. ISSN 0090-6778.

[Alderisi et al. 2015]ALDERISI, G. et al. Simulative assessments of the IEEE 802.15.4e DSME and TSCH in realistic process automation scenarios. In: *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*. [S.l.]: IEEE, 2015. p. 948–955. ISBN 978-1-4799-6649-3.

[Baccour et al. 2012]BACCOUR, N. et al. Radio link quality estimation in wireless sensor networks. *ACM Transactions on Sensor Networks*, v. 8, n. 4, p. 1–33, sep 2012. ISSN 15504859.

[Baronti et al. 2007]BARONTI, P. et al. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. *Computer Communications*, v. 30, n. 7, p. 1655–1695, may 2007. ISSN 01403664.

[Barrenetxea et al. 2008]BARRENETXEA, G. et al. The hitchhiker's guide to successful wireless sensor network deployments. In: *Proceedings of the 6th ACM conference on Embedded network sensor systems - SenSys '08*. New York, New York, USA: ACM Press, 2008. p. 43. ISBN 9781595939906.

[Bertocco, Gamba and Sona 2008]BERTOCCO, M.; GAMBA, G.; SONA, A. Is CS-MA/CA really efficient against interference in a wireless control system? An experimental answer. In: *2008 IEEE International Conference on Emerging Technologies and Factory Automation*. [S.l.]: IEEE, 2008. p. 885–892. ISBN 978-1-4244-1505-2.

[Cheffena 2012]CHEFFENA, M. Industrial wireless sensor networks: channel modeling and performance evaluation. *EURASIP Journal on Wireless Communications and Networking*, v. 2012, n. 1, p. 1–8, 2012. ISSN 1687-1499.

[Dargie and Poellabauer 2010]DARGIE, W.; POELLABAUER, C. *Fundamentals of Wireless Sensor Networks*. [S.l.]: John Wiley & Sons, Ltd, 2010. ISBN 9780470666388.

[De Guglielmo, Brienza and Anastasi 2016]De Guglielmo, D.; BRIENZA, S.; ANASTASI, G. A Model-based Beacon Scheduling algorithm for IEEE 802.15.4e TSCH networks. In: *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. [S.l.]: IEEE, 2016. p. 1–9. ISBN 978-1-5090-2185-7.

[Delgado Gomes et al. 2013]Delgado Gomes, R. et al. On the Impact of Local Processing for Motor Monitoring Systems in Industrial Environments Using Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, v. 2013, p. 1–14, 2013. ISSN 1550-1329.

[Du, Navarro and Mieyeville 2015]DU, W.; NAVARRO, D.; MIEYEVILLE, F. Performance evaluation of IEEE 802.15.4 sensor networks in industrial applications. *International Journal of Communication Systems*, v. 28, n. 10, p. 1657–1674, jul 2015. ISSN 10745351.

[Gomes et al. 2017]GOMES, R. D. et al. A Simulation Model for Industrial Multi-Channel Wireless Sensor Networks. *Journal of Communication and Information Systems*, v. 32, n. 1, p. 29–40, 2017. ISSN 19806604.

[Gomes et al. 2017]GOMES, R. D. et al. Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes. *Ad Hoc Networks*, feb 2017. ISSN 15708705.

[Gunatilaka, Sha and Lu 2017]GUNATILAKA, D.; SHA, M.; LU, C. Impacts of channel selection on industrial wireless sensor-actuator networks. In: *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. [S.l.]: IEEE, 2017. p. 1–9. ISBN 978-1-5090-5336-0.

[Gursu et al. 2016]GURSU, M. et al. Reliable co-existence of 802.15.4e TSCH-based WSN and Wi-Fi in an aircraft cabin. In: *2016 IEEE International Conference on Communications Workshops (ICC)*. [S.l.]: IEEE, 2016. p. 663–668. ISBN 978-1-5090-0448-5.

[Hahm et al. 2015]HAHM, O. et al. Operating Systems for Low-End Devices in the Internet of Things: a Survey. *IEEE Internet of Things Journal*, 2015. ISSN 2327-4662.

[Huang, Pang and Hung 2009]HUANG, Y.-K.; PANG, A.-C.; HUNG, H.-N. A comprehensive analysis of low-power operation for beacon-enabled IEEE 802.15.4 wireless networks. *IEEE Transactions on Wireless Communications*, v. 8, n. 11, p. 5601–5611, nov 2009. ISSN 1536-1276.

[Incel 2011]INCEL, O. D. A survey on multi-channel communication in wireless sensor networks. *Computer Networks*, v. 55, n. 13, p. 3081–3099, sep 2011. ISSN 13891286.

[Kazmierkowski 2014]KAZMIERKOWSKI, M. P. Industrial Communication Technology Handbook, 2nd Ed. [Book News]. *IEEE Industrial Electronics Magazine*, v. 8, n. 2, p. 67–68, jun 2014. ISSN 1932-4529.

[Khanafer, Guennoun and Mouftah 2014]KHANAFER, M.; GUENNOUN, M.; MOUFTAH, H. T. A Survey of Beacon-Enabled IEEE 802.15.4 MAC Protocols in Wireless Sensor Networks. *IEEE Communications Surveys & Tutorials*, v. 16, n. 2, p. 856–876, jan 2014. ISSN 1553-877X.

[Kotsiou et al. 2017]KOTSIOU, V. et al. LABeL: Link-based Adaptive BLacklisting Technique for 6TiSCH Wireless Industrial Networks. In: *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems - MSWiM '17*. New York, New York, USA: ACM Press, 2017. p. 25–33. ISBN 9781450351621.

[Kotsiou et al. 2019]KOTSIOU, V. et al. Blacklisting-Based Channel Hopping Approaches in Low-Power and Lossy Networks. *IEEE Communications Magazine*, v. 57, n. 2, p. 48–53, feb 2019. ISSN 0163-6804.

[Kusy et al. 2011]Kusy, B. et al. Radio diversity for reliable communication in wsns. In: *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*. [S.l.: s.n.], 2011. p. 270–281.

[Queiroz et al. 2017]QUEIROZ, D. V. et al. Survey and systematic mapping of industrial Wireless Sensor Networks. *Journal of Network and Computer Applications*, v. 97, p. 96–125, nov 2017. ISSN 10848045.

[Queiroz, Gomes and Benavente-Peces 2017]QUEIROZ, D. V.; GOMES, R. D.; BENAVENTE-PECES, C. Performance Evaluation of Default Active Message Layer (AM) and TKN15.4 Protocol Stack in TinyOS 2.1.2. In: *Proceedings of the 6th International Conference on Sensor Networks*. [S.l.]: SCITEPRESS - Science and Technology Publications, 2017. p. 69–79. ISBN 978-989-758-211-0.

[Sohraby, Minoli and Znati 2007]SOHRABY, K.; MINOLI, D.; ZNATI, T. *Wireless Sensor Networks: Technology, Protocols, and Applications*. [S.l.]: Wiley, 2007. ISBN 978-0-471-74300-2.

[Soua and Minet 2015]SOUA, R.; MINET, P. Multichannel assignment protocols in wireless sensor networks: A comprehensive survey. *Pervasive and Mobile Computing*, v. 16, p. 2–21, jan 2015. ISSN 15741192.

[Tanghe et al. 2008]TANGHE, E. et al. The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 MHz. *IEEE Transactions on Wireless Communications*, v. 7, n. 7, p. 2740–2751, jul 2008. ISSN 1536-1276.

[Tennina et al. 2013]TENNINA, S. et al. *IEEE 802.15.4 and ZigBee as Enabling Technologies for Low-Power Wireless Systems with Quality-of-Service Constraints*. [S.l.]: Springer Berlin Heidelberg, 2013. (SpringerBriefs in Electrical and Computer Engineering). ISBN 978-3-642-37367-1.

[Wang and Yao 2013]WANG, N.-C.; YAO, K. Characterizing fading channel under abrupt temporal variations. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. [S.l.]: IEEE, 2013. p. 5056–5060. ISBN 978-1-4799-0356-6.

[Watteyne et al. 2012]WATTEYNE, T. et al. OpenWSN: a standards-based low-power wireless development environment. *Transactions on Emerging Telecommunications Technologies*, v. 23, n. 5, p. 480–493, aug 2012. ISSN 21613915.

[Wehrle, Günes and Gross 2010]WEHRLE, K.; GÜNES, M.; GROSS, J. *Modeling and Tools for Network Simulation*. [S.l.]: Springer Berlin Heidelberg, 2010. ISBN 978-3-642-12330-6.

[Will, Schleiser and Schiller 2009]WILL, H.; SCHLEISER, K.; SCHILLER, J. A real-time kernel for wireless sensor networks employed in rescue scenarios. In: *2009 IEEE 34th Conference on Local Computer Networks*. [S.l.]: IEEE, 2009. p. 834–841. ISBN 978-1-4244-4488-5.

[Zadeh 1965]ZADEH, L. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, jun 1965. ISSN 00199958.

[Zhan, Xia and Anwar 2016]ZHAN, Y.; XIA, Y.; ANWAR, M. GTS size adaptation algorithm for IEEE 802.15.4 wireless networks. *Ad Hoc Networks*, v. 37, p. 486–498, feb 2016. ISSN 15708705.

APÊNDICE A – Fuzzy Logic Code in Python

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
\%matplotlib inline
import matplotlib.pyplot as plt

# New Antecedent/Consequent objects hold universe variables
# and membership functions

# Considering range from -40dBm, -70dBm e -100dBm and taking account
# -70dBm as default, -40dBm is 42% better than the default and
# -100dBm is 42% worse than the default
prr = ctrl.Antecedent(np.arange(0, 101, 1), 'PDR (%)') # 0 to 100%
rssi = ctrl.Antecedent(np.arange(-42, 43, 1), 'Variarion of RSSI (%)')
        # 0 to 100%
duplicate = ctrl.Antecedent(np.arange(0, 101, 1), 'Duplicate (%)')
        # 0 to 100%
channel_list = ctrl.Consequent(np.arange(0, 101, 1), 'Channel lists')
        # 0 to 100%

# Automatically creates mapping between crisp and fuzzy values
# using a standard membership function (triangle)
prr['bad'] = fuzz.gaussmf(prr.universe, 0, 18)
prr['acceptable'] = fuzz.gaussmf(prr.universe, 65, 8)
prr['great'] = fuzz.gaussmf(prr.universe, 100,10)

# Creates membership functions using different types
rssi['bad'] = fuzz.gaussmf(rssi.universe, -42, 18)
rssi['acceptable'] = fuzz.gaussmf(rssi.universe, 0, 5)
rssi['great'] = fuzz.gaussmf(rssi.universe, 42,18)
```

```
# Automatically creates mapping between crisp and fuzzy values
# using a standard membership function (triangle)
duplicate['suitable'] = fuzz.gaussmf(duplicate.universe, 0, 18)
duplicate['acceptable'] = fuzz.gaussmf(duplicate.universe, 40, 8)
duplicate['bad'] = fuzz.gaussmf(duplicate.universe, 100,25)

channel_list['Denylist'] = fuzz.gaussmf(prr.universe, 0, 16)
channel_list['Greylist'] = fuzz.gaussmf(prr.universe, 50, 8)
channel_list['Allowlist'] = fuzz.gaussmf(prr.universe, 100,16)

rule1 = ctrl.Rule(prr['great'] & (rssi['great'] | rssi['acceptable']),
        channel_list['Allowlist'])
rule2 = ctrl.Rule(prr['bad'] | rssi['bad'], channel_list['Denylist'])
rule3 = ctrl.Rule(prr['bad'] & rssi['acceptable'],
        channel_list['Denylist'])
rule4 = ctrl.Rule((prr['great'] | prr['acceptable']) &
        (rssi['bad'] | rssi['acceptable']), channel_list['Greylist'])
rule5 = ctrl.Rule(prr['acceptable'] & (rssi['acceptable'] |
        rssi['great']), channel_list['Allowlist'])
rule6 = ctrl.Rule(((prr['acceptable'] | prr['great'] | prr['bad']) &
        duplicate['bad']), channel_list['Denylist'])
rule7 = ctrl.Rule(duplicate['acceptable'] & (rssi['acceptable'] |
        rssi['great']), channel_list['Greylist'])
rule8 = ctrl.Rule((duplicate['suitable'] & prr['bad']),
        channel_list['Allowlist'])

lists_ctrl = ctrl.ControlSystem([rule1, rule2, rule3, rule4,
        rule5, rule6, rule7, rule8])
lists_simulator = ctrl.ControlSystemSimulation(lists_ctrl)

# Entering some values
lists_simulator.input['PDR (\%)'] = 95
lists_simulator.input['Variarion of RSSI (\%)'] = 3
lists_simulator.input['Duplicate (\%)'] = 15

# Computing the result
lists_simulator.compute()
print(lists_simulator.output['Channel lists'])
```