Analysis and Design of Control Plane Functions for Industrial Internet of Things

Dissertation

zur Erlangung des Doktorgrades an der
Fakultät für Angewandte Informatik
der Universität Augsburg

vorgelegt von

Dario Fanucchi

2021

Erstgutachter:	Prof. Dr.-Ing. Rudi Knorr
Zweitgutachter:	Prof. Dr. Jörg Hähner
Tag der mündlichen Prüfung:	17. September 2021

Abstract

With current low-power wireless standards, the mistrust about wireless technology for industrial applications is unjustifiable. For instance, a Medium Access Control (MAC) technique, called Time Slotted Channel Hopping (TSCH), has made wire-like end-to-end reliability, certified security, and over a decade of battery lifetime a reality in Industrial Wireless Sensor Networks (IWSNs). TSCH is integrated into the IEEE 802.15.4 standard, and it is a cornerstone of the open standardised protocol suite for the Industrial Internet of Thing (IIoT) proposed by IETF. Specifically, the IETF 6TiSCH stack combines the industrial performance of TSCH with a set of higher layers protocols providing IPv6-connectivity to constrained devices. Thus, it promises, above all, interoperability between vendors and seamlessly integration of IWSNs into the Internet.

Despite these high potentials and the high reputation of 6TiSCH in industry and academia, challenges remain, and some of its limitations need to be understood. In this thesis, we focus on the issues related to the harmonisation of the asynchronous IPv6-based upper layers with the synchronous TSCH technique, which rely on control plane primitives such as the network bootstrap procedure, the management of communication resources and the collection of network statistics. We identify in which circumstances the 6TiSCH standardised control primitives, which should lay the foundations for a reliable operational 6TiSCH network, exhibit limitations. After explaining the shortcomings and their causes, we design refinements and validate them simulatively. The focus is not on data transmissions but on mechanisms for a dependable exchanging of control messages. Nevertheless, these have been designed without significantly reducing the available bandwidth for data applications and the lifetime of power-constrained nodes. Accordingly, we provide the following main contributions.

First, we analyse the interplay between the scheduling of control messages and the multi-hop route computation during the network bootstrap phase, pointing out the limits of the current guidelines that may preclude or penalise the 6TiSCH network's operational state. Indeed, an improper choice of the protocol parameters may lead to a very long and energy-consuming network formation and stabilisation phase (e.g. more than 30 minutes even in small 5x5 grid networks).

Second, we examine different resource allocation strategies for bootstrapping a 6TiSCH network. Here, we design, implement and evaluate a scheduling mechanism

for coordinating the transmission of control messages among neighbouring nodes in a dynamic and distributed way. This mechanism has exhibited a significantly faster network formation than the default configuration, even in challenging chain network topologies, where it consumes at most $0.4\,\%$ of the battery's charge of a sensor node.

Finally, we investigate how to obtain an accurate Link Quality Estimation (LQE) in 6TiSCH. We demonstrate that state-of-the-art strategies, which are not designed having TSCH in mind, are too inaccurate for guaranteeing a reliable and stable 6TiSCH network setup. Indeed, internal interference hampers their link measurements. To overcome this issue, we propose a LQE strategy that allows a collision-free transmission of broadcast probe messages even during the network setup. This proposal improves the estimation accuracy dramatically, exhibiting a quite perfect estimation of at least $90\,\%$ of the links in different network topologies and in a short time (i.e. in order of minutes)

We are firmly convinced that 6TiSCH is an IIoT key enabler. Despite that, we forewarn the risk of its blind adoption as one-size-fits-all solutions in this work. Addressing some limitations in its control primitives and providing essential enhancements, we believe this thesis supports the future wide adoption of 6TiSCH in the industry.

Acknowledgement

During my work experience at the Chair for Communication System in the Department of Computer Science at the University of Augsburg, I met many people who made this thesis possible and help me to be the researcher I am now.

First and foremost, I would like to thank my supervisor Prof. Rudi Knorr for believing in my potential and for giving me the necessary freedom and constant support in my research.

Also, my sincere thanks go to Prof. Jörg Hähner for kindly accepting to serve as a second supervisor and for his interest in reviewing and evaluating this work.

I especially like to thank Prof. Barbara Staehle, who invested time and effort in guiding me to be on the right track. Her valuable suggestions and constructive comments have enriched the quality of this work.

Special thanks also go to Prof. Giuseppe Anastasi, Dr. Carlo Vallati and Dr. Francesca Righetti, for providing me with wise advice and engaging discussions during our collaboration in the last part of my research activities.

Thanks also to Neda, Erik, Ali, Henning, Michael, Josef and all other people I met at Fraunhofer ESK. I gained a lot from their expert knowledge. I am also grateful to Sabine, who gave me precious organisational support and, most important, friendly conversations that alleviate the stressful moments.

Thanks to my good friends in Augsburg and in Lucca. Thank you for remaining interested and supportive and giving me many things to enjoy outside my PhD journey.

Last but not least, profound gratitude goes to my family. To my parents and my sister for their belief in me and wholehearted support in my decisions. To my wife Giulia and my two daughters Adele and Bianca for being part of my life and pushing me to be a better person. I love you.

Dario Fanucchi,
Augsburg, June 2021

Contents

1 Introduction

1.1 Context

In the last two decades, the miniaturisation of computing, sensing and communication technologies has enabled the development of tiny, low-cost electronic boxes which can interact with their environment by sensing and/or controlling physical processes. These devices (*sensor nodes*, *motes*) contain at least a micro-controller for computing, a sensing unit as an interface with the physical world, a radio for communicating with other devices, and are typically powered by a battery (see Section 2.2.1 for more details). When deployed in a particular area of interest, these sensor nodes form a **Wireless Sensor Network (WSN)**. Nodes are organised around one particular device, called *gateway* or *sink*. Typically, it is less constrained (e.g. mains powered, with a powerful processing unit, a lot of RAM and storage volume), collects the measurements and connects the WSN to other computer networks. As depicted in Figure 1.1, there are three general ways to connect the sensor nodes to the sink. In a WSN with a *star* topology, each sensor node communicates directly with the sink using a single hop. This approach is energy-efficient and straightforward but does not cope well with the short-range communication and the large geographic area that a WSN typically covers. Consequently, *multi-hop cluster-tree* or *mesh* networks are common topologies for WSNs. Some nodes are *cluster heads* (or *coordinators*) and also serve as *relays* for the other *leaf nodes* in the former ones. In the latter ones, all nodes act as *forwarders* and collaborate to propagate sensor data towards the gateway.

WSNs can not rely on communication protocols and algorithms proposed for *Mobile Ad Hoc Networks (MANETs)* and traditional *Wireless Local Access Networks (WLANs)*. This is principally due to (i) the low data rates involved and (ii) the constraints in terms of processing, memory and energy of sensor nodes. Thus, specific low power wireless networking technologies have been designed for making WSNs reliable, energy-efficient, scalable, adaptable and easy to deploy. Chapter 2 provides insight into how and to which degree those goals are achieved today.

As a result, WSNs are generally recognised as key elements for paving the way to the realisation of the **Internet of Things (IoTs)** vision [1]. To this end, it is still

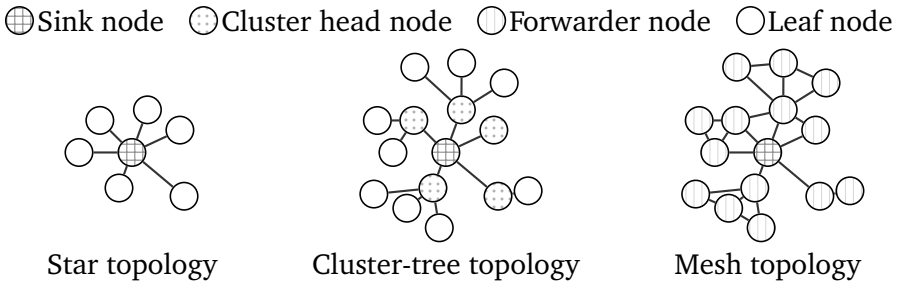


Fig. 1.1: Different WSN topologies

necessary to integrate WSNs, or even each of their device, in the Internet and moving away from often disconnected WSN islands, usually based on semi-proprietary, not interoperable protocols and architectures. This aspect is particularly important for industrial applications. Here, the lack of regulation has produced numerous technical and commercial success stories of **Industrial Wireless Sensor Networks (IWSNs)**, which are based on similar, but not compatible, proprietary solutions. Further, seamless integration into the Internet magnifies the potential of isolated industrial plants by augmenting their flexibility, their manageability, and disclosing the opportunity to deploy new services [2].

As we explain later in Section 2.4.1, two of the most influential Standard Development Organisations (SDOs), namely the Institute of Electrical and Electronics Engineering (IEEE-SA) [3] and Internet Engineering Task Force (IETF), are now promoting the adoption of the *IPv6 over the TSCH mode of IEEE 802.15.4 (6TiSCH)* for the **Industrial Internet of Thing (IIoT)**. 6TiSCH is a complete networking stack. It is based on the industrial low-power wireless technology Time Slotted Channel Hopping (TSCH) at the bottom and on lightweight IPv6-enabled higher-layer protocols, i.e. IPv6 over Low power Wireless Personal Area Network (6LoWPAN), Routing Protocol for Low Power and Lossy Networks (RPL), Constrained Application Protocol (CoAP). Thus, 6TiSCH ensures both dependable wireless communications (i.e. accomplishing strict requirements in terms of reliability, delay and security [4]) and the capacity, for industrial machines, to communicate with each other device and/or the Internet.

At the time of writing, the 6TiSCH protocol stack is attracting attention in industry and academia, and a very large adoption in various key IIoT areas seems reasonable. Firstly, it targets applications in *condition monitoring* and *process automation* domains. Condition monitoring deals with diagnostic and maintenance of structures (buildings, plants, bridges, etc.) or machine equipment. Instead, process automation

Tab. 1.1: Different classes of industrial applications as defined by ISA [5] and [7]

Category	Class	Application
Safety	0	Emergency action (always critical)
Control	1	Closed-loop regulatory control (often critical) function
	2	Closed-loop supervisitory control (usually non critical)
	3	Open-loop control (human in the loop)
Monitoring	4	Alerting (short-term operational consequence)
	5	Logging (no immediate operational consequence)

provides information regarding the resources (i.e. materials, supply chain status and workforce) for the production and service provision. Examples of process automation industries are oil and gas distribution, electric power generation and management, chemical processing and mineral treating. Considering the classification defined by International Society of Automation (ISA) [5], in which the criticality of applications is considered (see Table 1.1), these two domains reside from Class 2 to 5. Those applications demand the industrial requirements listed in Table 1.2. When data is lost or comes too late, the consequences are severe (e.g. explosions, material losses and interruption of continuous industrial processes). However, TSCH-based technologies can meet these strict requirements nowadays. Two other exciting applications, which we expect 6TiSCH can cover soon, are intra-vehicle and intra-avionics wireless networks [6]. Nevertheless, smart home, smart city and smart agriculture, which are less critical IoT application domains, could also greatly benefit from adopting the 6TiSCH protocol stack.

All the above things considered, we concentrate on the 6TiSCH protocol stack in our research, and we investigate the remaining challenges that may delay or hinder its wide adoption. We particularly focus on the unresolved issues related to the harmonisation of the IPv6-based upper layers with the TSCH technique.

Tab. 1.2: Industrial applications and their typical requirements (source: [8], [9])

	Condition Monitoring	Process Automation
Reliabilty	$1 - 10^{-5}$	$1 - 10^{-5}$
Latency	100 ms	50 ms to X s
Data rate	kbit s^{-1}	kbit s^{-1}
Link distance	0.1 km to 1 km	0.1 km to 1 km
Node density	10 m^{-2} to 20 m^{-2}	10 m^{-2} to 20 m^{-2}
Lifetime	3 Years	> 5 Years

1.2 Problem Statement and Contributions

As mentioned above, 6TiSCH has been receiving plenty of interest in academia and industry in the last years. In particular, we observe numerous research efforts addressing the scheduling problem in TSCH, i.e. how a scheduler assigns transmission resources among the different nodes in the network (see Section 2.4.5). Most of these proposals have in common the focus on reliable and predictable data transmission and the assumption that the 6TiSCH network is already operational, i.e. nodes are ready to send data messages. Consequently, it is common to consider a few dozen minutes (as allowance time) for the network to form and stabilise, before the actual performance evaluation of the scheduling algorithm starts. What happens in this meantime is usually out of interest for the 6TiSCH community or implementers of IIoT solutions.

In contrast, our research work puts the light on this crucial network bootstrap phase. During this phase, several control plane primitives take the stage: *network advertising, joining, synchronisation, neighbour discovery, topology establishment*. As we explain later in Chapter 2, the IEEE 802.15.4 standards and other IETF standardisation efforts specify basic implementations of these control plane mechanisms. These aim to realise a *minimal mode of operation* to mostly guarantee interoperability between nodes in a 6TiSCH network. Therefore, it comes as no surprise that some aspects need to be optimised or improved. For instance, default solutions for the TSCH network synchronisation assume that joining nodes have their radio always on, while scanning the available channels for network advertising messages. Thus, minimising the network joining time in each node allows significant energy savings. Another example concerns the topology establishment/control. Both the Medium Access Control (MAC) and the Network (NET) layers build a multi-hop topology in parallel: while each node chooses which neighbour nodes to communicate with at MAC layer, at NET layer parent-child relationships are established. As we highlight in this thesis, there are interactions between these two parallel processes that need to be regulated for avoiding an unsuccessful 6TiSCH network formation or inefficient performance at runtime. Finally, Link Quality Estimation (LQE) is another crucial but also not trivial control primitive in 6TiSCH. Both routing and scheduling algorithms rely on accurate link metrics for selecting the best paths and allocating enough transmission opportunities to each node during the 6TiSCH network bootstrap phase. Unfortunately, state-of-the-art LQE strategies intended for WSNs adopting a Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme do not perform in the same way in the time-slotted and multi-channel environment defined

by TSCH. Here, a proper LQE mechanism should consider the network schedule and differentiate the link measurements taken at diverse kinds of TSCH links.

Accordingly, the research presented in this thesis will provide the following main contributions. The first contribution is to analyse the interplay between the scheduling of control messages and the multi-hop route selection during the 6TiSCH network formation phase. We point out the limitations of the current guidelines that may preclude or penalise the 6TiSCH network's operational state.

The second contribution is to examine different resource allocation strategies for bootstrapping a 6TiSCH network. Here, we design, implement and evaluate a scheduling mechanism for coordinating the transmission of control messages among neighbouring nodes in a dynamic and distributed way.

The third contribution is to investigate how to obtain an accurate LQE in 6TiSCH. We demonstrate that state-of-the-art strategies, which are not designed having TSCH in mind, are too inaccurate for guaranteeing a reliable and stable 6TiSCH network setup. To overcome this issue, we propose a strategy that allows a collision-free transmission of broadcast probe messages even during the network setup.

1.3 Methodology and Assumptions

As already mentioned, our intention at the outset of this research are, firstly, to *identify* in which circumstances the 6TiSCH standardised control primitives exhibit limitations; secondly to *design and implement* enhancements for a reliable bootstrapping process and an accurate link quality estimation in 6TiSCH networks; thirdly to *validate* the benefit of our proposals.

Contiki [10], its fork Contiki-NG [11] and the included Cooja framework play a central role in accomplishing these research intentions. We describe their characteristics in more details in Section 2.2.3. In short, they are well-acknowledged tools for WSNresearch and are widely adopted by academia and industry for protocol prototyping. Since 2015, Contiki supports the 6TiSCH stack, with an implementation tested at several interoperability events related to the standardisation activities of 6TiSCH. The Cooja simulator allowed us to extensively evaluate the performance of both state-of-the-art (or standardised) solutions and our proposed enhancements. Thus, we could consider several different scenarios and investigate the impact on the performance of several protocol or algorithm parameters without the influence of external side-effects. Moreover, for the investigation on LQE within 6TiSCH, Cooja

has allowed us to set the nominal characteristics of each link for easily evaluating the estimation error of different strategies under the same conditions. At the same time, we have implemented in Contiki our algorithms proposed in this thesis, before validating them with Cooja. Thus, the same code can be flashed and run on a device such as Zolertia RE-Mote [12] for real experiments in a testbed. However, we have decided to do without this further step (i.e. measurements in a testbed under real-world conditions) after taking cost-benefit aspects into consideration.

For the investigation on the interplay between the MAC and NET layers during the 6TiSCH network formation process in Chapter 4, we have moved from analytical methods, using Discrete Time Markov Chain (DTMC) and probability formulas to model the behaviour of nodes and the diffusion of control messages in this phase. However, it was for us complicated to obtain tractable mathematical expressions without introducing some simplifying assumptions, which distort our proposed solutions. Therefore, we do not report on that in this thesis.

As expressed in Section 1.1, our research work addresses WSNs with a combination of strict requirements in terms of reliability, latency and energy, which can be satisfied through the 6TiSCH protocol stack. Following [7], we examined in this thesis 6TiSCH networks that are composed of no more than hundreds of nodes. Further, we assume that nodes are deployed following some rules in a building- or campus-size area. These assumptions are also motivated by our belief that larger networks or unplanned deployments are unrealistic for IIoT applications that demand both high-reliability, latency guarantee and years of battery lifetime.

The nodes are constrained devices in terms of processing, memory and energy resources (see Section 2.2.1). Despite the advancement in powerful micro-controller and lightweight machine learning algorithms, to the best of our knowledge, it is still not practicable the adoption of Artificial Intelligence (AI) (e.g. for LQE problem in 6TiSCH) on these kinds of nodes, especially in parallel with the actual monitoring or process application. Thus, the conceptual design of our proposals takes care of the processing and radio activities of nodes for avoiding a significant reduction of their lifetime.

1.4 Thesis Outline

To address the previously discussed contributions, this thesis structures its content as follows.

Chapter 2 present the fundamentals of low power wireless networking, characterising the constrained devices that form the network and describing the importance of the lower communication layers and their design challenges for making the network able to fulfil the strict reliability, latency and energy requirements of industrial IoT applications. Additionally, it introduces the 6TiSCH protocol stack and its key elements. Finally, an overview on other competing low-power standards for the (industrial) IoT is provided.

After providing the background knowledge, we position this thesis's contributions within the low-power wireless networking and 6TiSCH community in **Chapter 3**, where we review related works both on bootstrapping procedures and on LQE mechanisms for low-power multi-hop WSNs. Here, we highlight the difference between our proposal and the major existing solutions proposed so far.

In **Chapter 4**, we investigate the dynamic of the 6TiSCH network formation process. This chapter first highlights how the blind adoption of the standardised procedure proposed by 6TiSCH WG produces a very long (or even unsuccessful) network formation phase in dense network topologies. Then, the chapter describes Multiple Advertisement Cell Allocation (MACA), our proposed dynamic allocation strategy for improving the 6TiSCH network formation.

Chapter 5 deals with the LQE mechanism in 6TiSCH network. It investigates the performance of popular LQE strategies not designed having TSCH in mind, and it demonstrates that they do not perform well on 6TiSCH networks, especially during the crucial initial phase. Additionally, this chapter introduces Dedicated Broadcast Slotframe for Probing (DBS-P), our proposed LQE approach for improving the accuracy of Expected Transmission Count (ETX) estimations and, thus, better supporting routing and scheduling during the 6TiSCH network setup.

Finally, **Chapter 6** summarises this thesis. Additionally, it points out some perspectives of future research directions.

2 Background on Wireless Sensor Networking for the Industrial IoT

2.1 Chapter Overview

This chapter establishes the background knowledge for the contributions proposed in the rest of this work.

First, we focus on the Internet of Thing (IoT) device, on which the algorithms and mechanisms presented in this thesis run. We describe its typical hardware architecture and its limitations in terms of computing, memory and power resources. Moreover, we point out the role of the software in these devices; it must carefully manage the limited resources while providing an essential and flexible programming interface for developers across heterogeneous hardware platforms and application scenarios. To explain that, we concisely describe Contiki, the embedded Operative System (OS) that we selected to implement and evaluate our research.

Secondly, we focus on the fundamentals of low power wireless networking, considering the lowest three layers of the Open System Interconnection (OSI) model [13], i.e. the Physical (PHY), the Medium Access Control (MAC) and the Network (NET) layers. First, we give a concise description of low power wireless links, explaining the reasons for their unreliability and time-variance. Then, we point out the importance of the MAC layer and its design challenges for making a Wireless Sensor Network (WSN) dependable, i.e. able to fulfil the strict reliability, latency and energy requirements of Industrial Internet of Thing (IIoT) applications. Finally, we present the significant design challenges and the suitable approaches for routing in Industrial Wireless Sensor Networks (IWSNs). The suitability of gradient routing and the necessity of an efficient, accurate and adaptive link-quality estimation are highlighted.

Thirdly, we focus our attention to the several low-power wireless standards and technologies for IIoT. We explain how the cooperation between two prominent standardisation bodies, namely the Institute of Electrical and Electronics Engineering

(IEEE-SA) [3] and Internet Engineering Task Force (IETF) [14], has produced a promising set of protocols to bring IPv6 connectivity to a proven industrial low-power wireless techonology. We detail on several key elements of this 6TiSCH protocol stack, namely MAC protocol Time Slotted Channel Hopping (TSCH), the routing protocol Routing Protocol for Low Power and Lossy Networks (RPL) and the 6TiSCH management plane. The latter, together with the interplay between TSCH and RPL protocol, are central topics in this thesis.

2.2 Characterisation of Constrained Devices

A premise of (I)WSNs is that the sensor devices, which form the network, have typically a limited supply of resources for sensing, processing and communicating data. The IETF coins the term *constrained device* [15]. Nevertheless, *WSN platform*, *low-end IoT device*, *sensor/actuator*, *smart object* are others term often used in this context.

In this section, we focus on such devices, i.e. microcontroller-based boards with a small size and too resource-constrained to run traditional OS such as Linux or Windows 10 IoT Core [16], [17]. First, we describe the hardware architecture of a constrained device and the functionalities of the different subsystems, which form it. Moreover, we provide examples of state-of-the art WSN platforms that can run the techniques and algorithms object of this thesis. Then, we briefly describe the software that run on WSN devices and provide basic programming abstractions to application developers. Finally, we introduce Contiki, the OS for WSN that we have selected for implementing and evaluating our contributions within this thesis.

2.2.1 Node Architecture

From a hardware point of view, a generic WSN platform consists of four main components, as shown in Figure 2.1, the power unit, the sensing unit, the processing unit and the communication unit.

The **power unit** supplies power to all active components in the node's platform. Basically, it consists of a battery and a DC-DC converter. The battery is exhaustible. The rate at which a battery discharges (without significantly affecting the prescribed supply voltage) is expressed in milliamperes per hours ($\mathrm{mA\,h}$)[1] by its capacity C.

[1]The SI system defines the coulomb as the unit of electric charge in terms of the ampere and second: $1\,\mathrm{C} = 1\,\mathrm{A} \times 1\,\mathrm{s}$, hence $1\,\mathrm{mA\,h} = 3.6\,\mathrm{C}$

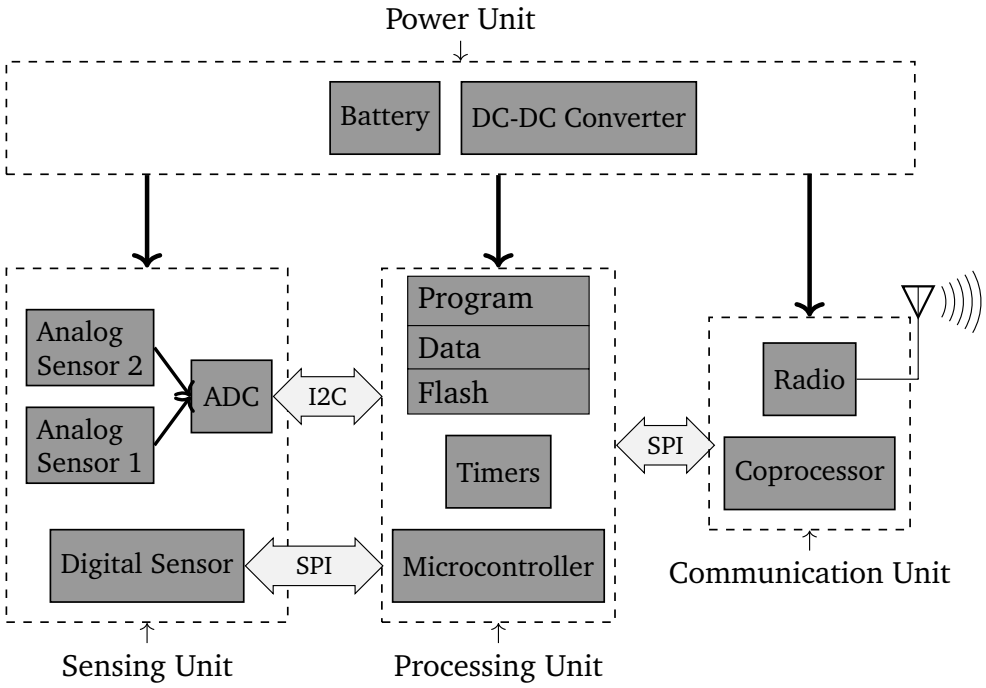


Fig. 2.1: Typical architecture of a WSN platforms

Thus, the capacity denotes the provided current over time. For example, a lithium AA battery such as [18] provides about $3000\,\mathrm{mA\,h}$ with a nominal voltage of $1.5\,\mathrm{V}$. It means, such a battery can, in theory, supply a device for $100\,\mathrm{h}$ (ca. 4 days), when the system takes on average $30\,\mathrm{mA}$. Therefore, to make a node last for a long time (i.e. years or a decade) on its battery, the average current draw of its hardware should range from $30\,\mu\mathrm{A}$ to $0.3\,\mathrm{mA}$. These numbers may be achieved only by adopting low power microcontrollers, by optimising the communication protocol and by carefully implementing the interactions between hardware and the OS architecture [19]. Alternatively, the power unit may recharge the battery by harvesting energy from the surrounding environment. In this case, potential energy sources are radiant sources (e.g. solar, RF waves), mechanical sources (e.g. wind, vibration) and thermal sources (e.g. external heat) [20]. The DC-DC converter transforms the voltage supplied by the battery to the specific voltage level required by each individual sub-circuit in the node's hardware.

The **sensing unit** is the interface of the device with the physical world. Thanks to Microelectromechanical System (MEMS) technology, it may integrate, into a small volume and at a low price, several types of sensors, such as weather sensors (wind, humidity, temperature), motion sensors (accelerometers, gyroscopes, magnetometers), pressure sensors, vision sensors (cameras, light, reflection), proximity sensors, microphone, moisture sensors, etc. Sensors convert the measured physical attributes

of the node's environment into an analogue signal (typically into electric current or voltage). Then, an Analog-to-Digital Converter (ADC) uses signal processing for transforming the output of the sensor into a digital signal. The sensor package itself (i.e. a digital sensor) or a configurable analogue-to-digital subsystem in the processing unit may perform ADC operations.

The **processing unit** of a WSN node consists of at least a processor chip, a non-volatile memory (i.e. ROM, EEPROM or flash) for storing program instructions, an active memory for temporarily storing the sensed data or doing computation (i.e. Random Access Memory (RAM)) and various peripheral interfaces such as I/O lines and timers. For certain specific tasks (e.g. encryption), this subsystem may also contain a custom logic. The processor coordinates the functions of a constrained device, executing the firmware, which contains the networking software and the communication interfaces with the sensing unit. Typically, a low power Microcontroller (MCU) is adopted. This MCU has multiple operational states, each of them characterised by a specific clock source/frequency and a different subset of active peripherals. When the MCU is in the active state, the system clock frequency is in the range of tens ofMHz, and there are no functional limitations. However, to lower its power consumption, the MCU spends the vast majority of its time in one of the various sleep modes, it quickly wakes up on an event and returns to sleep after making simple computations. Often, in the deep-sleep state, everything is turned off except a $32\,\mathrm{kHz}$ low-power slow clock, which allows the system to maintain a timer base by producing a current consumption in the range of some $\mu\mathrm{A}$ [21]. The amount of memory on a typical WSN node is still measured in kilobytes or few megabytes. For instance, the OpenMote-B [22], a very popular platform in the low-power wireless research community at the time of writing, comes with $512\,\mathrm{kB}$ for program storage and $32\,\mathrm{kB}$ of RAM. Besides, Direct Memory Access (DMA) has been widely used in the processing unit for making the I/O data transfer between memory and peripherals without using costly CPU resources [21].

The processing unit interacts with components outsides through several peripheral communication methods (i.e. **peripheral interfaces**). Widely featured interfaces are General Purpose Input Output (GPIO), Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C), Universal Asynchronous Receiver-Transmitter (UART) and the Universal Serial Bus (USB). For instance, the processor unit typically interacts with the communication unit using an SPI bus and GPIO lines, some of which are interrupt-capable.

The **communication unit** consists of a radio transceiver, which connects the sensor node with other devices in the network. Typically, the radio chip utilises the sub-GHz

Tab. 2.1: Current consumption during different radio states

	Sleep	Tx (0 dBm)	Rx	Listen
TI CC2420	426 µA	17.4 mA	18.8 mA	18.8 mA
LTP5800	11.2 µA	5.4 mA	4.5 mA	4.5 mA
TI CC2538	1.3 µA	24 mA	20 mA	24 mA
TI CC2650	1 µA	6.1 mA	5.9 mA	5.9 mA

or 2.4 GHz Industrial Scientific and Medical (ISM) frequency bands, since they do not require a license. The choice between one of those two bands depends on the application requirements and criteria such as range, data rates, antenna size, power consumption and support of the communication protocol (c.f. Section 2.5).

The communication unit is a significant power-hungry component in a WSN node. Table 2.1 presents the current drain under various operation modes of several radios widely adopted in the WSN platforms. We note that the typical current consumption is in the 5 mA–30 mA range for 2.4 GHz IEEE 802.15.4 communications, regardless of whether the radio is transmitting or receiving a frame. Besides, we observe that idle listening (i.e. keeping unnecessarily the radio active on an idle channel) has a comparable current drawn. From these numbers, we deduce that the radio transceiver should be active only for a tiny fraction of the time (e.g., 5 %, 1 % or less) to make battery powered devices last years or a decade. As we explain later in Section 2.3.2, this contraint requires the design of WSN-specific MAC protocols, which turn off the communication unit most of the time and activate it only when necessary or periodically.

Considering the relevant sensor system platforms over the past 20 years, these range from decade-old two-chip solutions (with one main MCU and a separate radio chip) to modern System on a Chip (SoC) solutions, which nowadays dominate the market. The former ones adopt ultra-low-power 8-bit or 16-bit microprocessors, such as Atmel ATmega128L or Texas Instruments (TI) MSP430, respectively. In particular, these two MCU were oft combined with the TI CC2420 radio unit, such as in the MicaZ [23], TelosB [24] and Z1 [25]. In contrast, modern SoC solutions integrate a 32-bit MCU, 2.4 GHz-radio and ample memory resources (e.g. 512 kB Flash, 32 kB RAM) in a single unit. In this context, the ARM Cortex-M3 MCU is the embedded architecture adopted by the majority of sensor platform available on the market, e.g. LTP590x mote modules [26], OpenMote [27], RE-Mote [12] and I3Mote [28]. When the OS is carefully optimised, such solutions may deliver the extremely low idle current consumption achieved by 16-bit platforms [29].

2.2.2 Operating Systems for Constrained Devices

We mentioned earlier the severe resource constraints in terms of processing power, memory size and energy of WSN node platforms. In parallel, there is a wide variety of hardware and communication technologies for different application scenarios. Both facts do not allow either to install a traditional OS or an application-specific, single plain C file on wireless sensor nodes. Instead, an OS for constrained devices (or embedded OS) is required. Essentially, it provides a basic system programming interface for the application developers, facilitating large scale software development, deployment and maintenance [16]. In doing that, an embedded OS does not merely achieve a low memory footprint (in the order of tens of kB of RAM and hundreds of kB of ROM), but address other several crucial issues, as exhaustively discussed in [16], [30], [31]. In a nutshell, the following two features determine the success and applicability of an OS in this context:

- *Efficiency in resource allocation*: An OS for constrained devices must be able to use efficiently the limited resources which are available for running IoT applications without issues. First, the type of the scheduler (i.e. *pre-emptive* or *cooperative*) and the execution model (i.e. *event-driven* or *multi-threaded*) affects the real-time capabilities and the efficiency of the system. Next, the code size and the RAM usage of the OS must be as compact as possible; indeed, the OS has to fit within the few kilobytes of memory provided with WSN platforms and to leave room for the application running on top of it. Finally, the effective power consumption of WSN devices substantially depends on (i) how the OS controls the MCU, and (ii) how the driver implementation makes use of advanced hardware power management features today available [19].

- *Modularity*: Besides the basic system functionality provided by the kernel (e.g. task scheduling and memory management), several distinct lightweight modules should compose an OS for constrained devices. Examples of such modules are the layers of the communication protocol stack, sensor devices drivers or high-level functionality, e.g. estimation of energy consumption, interactive shell. Among other things, this modular approach in structuring the OS (i) reduces memory requirements (by compiling only the required modules for the specific use case), (ii) allows quickly reprogramming of nodes at run-time (by loading only individual services), and (iii) facilitates the support of heterogeneous hardware (by introducing a hardware abstraction layer).

Prominent examples of embedded OSs that provides the above features includes TinyOS [32], Contiki [10], RIOT [33], FreeRTOS [34], μC/OS-II [35]. We refer the

reader to [16] for a survey and detailed comparison of these embedded OS for WSN and low-end IoT devices.

As already explained in Section 1.3, we have selected Contiki to implement and evaluate our contributions within this dissertation. Next, we concisely describe this well-acknowledged OS for constrained devices.

2.2.3 Contiki and Contiki-NG

Contiki is an open-source, memory-efficient OS for embedded wireless devices that brings IPv6 connectivity for sensor networks. Started by Dunkels at SICS in 2002, the project Contiki has grown rapidly, involving developers both from research institutions and major industry players. Written in C and released under an open-source license, the Contiki code has been regularly updated, continually enlarging the support of the protocols standardised by IETF for low-power IPv6 connectivity in the IoT. Since 2017, the project **Contiki-NG** takes care of the open-source development and maintenance of Contiki, focusing principally on the IETF 6TiSCH stack and adding the support for modern IoT platforms.

A Contiki system relies on modular, service-based **architecture**, where both the core code with system libraries and the application code are combined into a platform-specific firmware image during compilation. The core code of Contiki includes system primitives such as processes and timers, two different communication stacks, and several libraries or services.

Contiki is characterised by **low system requirements**. For example, Contiki-NG requires approximately $38\,\mathrm{kB}$ of ROM and $11\,\mathrm{kB}$ of RAM for running a full IPv6 stack with 6LowPAN, RPL and their default configuration parameters [11].

Processes and protothreads are the essential building blocks of both Contiki's services and applications. A process in Contiki is a code block which performs a task when something happens and then suspends its execution, waiting to be invoked again by the process scheduler. As a programming abstraction, protothreads [2] are a simplified and lightweight (stackless) version of normal threads, designed for event-driven systems. They provide a *blocking and wait* semantics in the execution code of the process. In sum, Contiki results in a hybrid OS regarding the **execution paradigm**: it adopts an event-driven kernel, but it also supports cooperative multi-threading through protothreads.

[2] The concept of protothreads was developed within the context of Contiki. Still, it has found many applications as a general-purpose C-library

Concerning **networking**, Contiki provides two main communication stacks: (i) the uIP stack [36], which is a small IETF-compliant TCP/IP stack for both IPv4 and IPv6 connectivity, and (ii) the Rime stack [37], which is a set of lightweight communication primitives typically used by sensor network protocols. At MAC layer, Contiki supports multiple mechanisms such as Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA), low power listening, X-MAC, TSCH, Bluetooth Low Energy [38].

Contiki has been designed with portability in mind. It provides a sort of **hardware abstraction layer** by defining a set of functions for common hardware components (such as timers, GPIO, UART, SPI, I2C, ADC, LEDs, buttons, sensors, radio) that the architecture-specific device driver of a platform must implement. Thus, almost all popular WSN node platforms available on the market are supported.

A Contiki system includes a **network simulator** called Cooja, which allows a developer to test applications or protocols in a repeatable manner and with full control over the network condition. It is a Java-based application; its GUI allows, among others, to define the network topology, to set the radio propagation model, to show the serial line output of every node and to visualise the communication activities in a timeline. Further, Cooja embeds emulator software for executing cycle-exact emulation of hardware platforms, and debugging the source code in a step-by-step fashion, before burning it into real hardware. We adopt the Cooja simulator for validating our proposals in Chapters 4 and 5.

2.3 Fundamentals of PHY, MAC and NET layer

2.3.1 Characterization of Low-Power Links

In general, when a wireless channel serves as a physical medium for the communication, we deal with a link that has an inherent grade of unreliability. Indeed, the transmitted radio signal undergoes several physical phenomena while travelling, and there is always a degradation of the signal quality at the receiver. If the received signal strength is below a certain threshold, then the receiver can not correctly detect the signal. As extensively described in [39], the primary physical phenomena that may led to unsuccessful transmission over a wireless channel are path loss, shadowing, multipath fading and external interference.

These physical effects are even more significant in WSNs, where small transmission power and cheap hardware platforms are employed. Consequently, low-power

wireless links are more unreliable and time-variant than links in other wireless systems. Plenty of research efforts [40]–[42] have empirically proved this fact. For the sake of brevity, we limit us to list the following experimental key observations (which were selected from the literature) in order to give a concise description of low-power wireless links:

- "*The transmission range of a node is divided into three regions: (1) a connected region, which is characterized by links of good quality, (2) a large transitional region, where links are of intermediate and unpredictable quality and (3) a disconnected region, where only bad links are present.*" [43]. Both the used radio hardware platform and the environment make the extension and the shape of these regions irregular and time-variant.

- "*Link located in the connected and disconnected regions tend to be stable*" [44], but most of the intermediate links experience bursts over short periods instead, where either all or no packets are received [45]. The variability over time of the link quality is due to changes in the environment or caused by the use of different channels for communication.

- "*Links of intermediate quality tend to be asymmetric*" [40], i.e. there is a significant difference in term of connectivity between both directions of a given link (i.e. uplink and downlink). Moreover, the asymmetry-level changes across channels [46]. In the transitional region, even small variations both in the transmit power and in receiver sensitivity at the two ends of a link lead to the asymmetry [47]. However, for links with good or poor quality the symmetry has been observed [40].

- "*External interference from Wi-Fi networks or Bluetooth systems can cause losses at multiple nodes*" [42]. When a WSN adopts the 2.4 GHz band, external interference caused by other higher power networks (such as WiFi-Networks or Bluetooth systems) is very common and represents a major reason for link asymmetry [48].

In this context, it is important to obtain a statistical characterisation of the link reliability, i.e. Link Quality Estimation (LQE). We discuss in more detail this challenging task and different methods and strategies that have been prososed in the literature in Section 3.3.

2.3.2 The Importance of Designing the MAC Layer

In the research community, it is a well-known fact that the MAC layer tremendously impacts the overall performance of a WSN. Primarily, the MAC layer determines the node's and network's lifetime, since it regulates the activities of the radio, one of the most energy-consuming components in a sensor node. Besides, by deciding when a node accesses the medium and how to resolve any potential conflicts, the MAC layer indirectly determines the communication latency and reliability. Finally, the MAC layer defines the frame format and, therefore, the amount of data the nodes can produce pro each transmission activity. The use of traditional wireless MAC techniques such as CSMA/CA, Multiple Access Collision Avoidance (MACA) [49] or other derivative protocols is not practicable in WSNs, in particular for the following three reasons: (i) resource limitations (in term of processing, memory and energy) of typical sensor nodes' hardware, (ii) multi-hop network topology and (iii) different typical application requirements (especially in term of throughput) [50]. Note that energy efficiency is considered the most crucial attribute of MAC protocols in traditional WSNs; nevertheless, high end-to-end reliability and latency guarantee are also prime requirements to be fulfilled in the industrial IoT context.

There has been a tremendous amount of research on the design and implementation of (I)WSN-specific MAC protocols. The choice of the optimal MAC protocol is challenging, even for a given application scenario with specific requirements in terms of energy consumption, reliability, delay and throughput. For the sake of brevity, this section does not survey the multitude of MAC protocols for WSNs. The interested reader can refer to [51]–[54], i.e. a selection of high-quality survey works on this topic. In the following, we explain the specific requirements and problems a MAC protocol have to face in industrial WSNs. In doing that, we offer background information for a detailed explanation of the TSCH MAC technology, which is central in our research work (see Section 2.4.3).

1. *Node Deployments:* The deployment of sensor nodes determines the physical wireless connectivity between sensors in the coverage area, i.e. the network density and the number of disjoint multi-hop paths between any pair of sensors. It can be either *planned* or *random*. In the former, the network installation follows guidelines that are specified in agreement with the design and the features of the adopted MAC protocols. In contrast, a random deployment is more challenging for protocol designers, since the MAC mechanisms must ensure (i) scalability to different network density and (ii) adaptability to various network size and topology.

2. *Node Roles or Functionalities:* MAC protocols significantly differ in their design if the devices that form the WSN have the same hardware capabilities (i.e. battery, memory, processor) or not. In the latter situation (i.e. *heterogeneous network*), it is generally straightforward to (statically) assign network roles (e.g. sink, relay or leaf nodes) considering the different resource constraints of the devices. Instead, in a *homogeneous network*, the MAC protocol must feature mechanisms to dynamically assign or alternate the nodes' roles in the network (e.g. electing a leader to bootstrap and maintain the network [55]).

3. *Duty Cycling:* Recall from Section 2.2 that the radio transceiver should stay turned off most of the time to make battery-powered devices last a decade. Hence, a MAC protocol for WSN relies on a duty-cycling mechanism that lets each node periodically alternates between a sleep and an active state. In the latter one, transmitting, receiving and listening activities take place. The fraction of the time a node spends in the active state is called duty-cycle. A low duty-cycle leads to low energy usage; still, it produces high latency and also increases the likelihood of collisions [56]. As a consequence, setting the duty-cycle must consider the trade-off between energy consumption, reliability and delay imposed by the application requirements. A prerequisite for duty cycling is the coordination of the active state among neighbours. This coordination can be achieved through *time synchronisation* (see below), alternative *asynchronous schemes* such as *preamble sampling* (sometimes also referred to as *low power listening*) [57] or adding an auxiliary ultra low-power transceiver (*wake-up radio)* in the communication unit [58].

4. *Sources of energy inefficiency:* Energy-efficiency is a traditional design goal for WSN-specific MAC protocols. It consists of addressing the following sources of energy waste [51], [59]:

    - *Idle listening:* It describes a node having the radio transceiver on waiting in vain for a potential frame. That usually happens when nodes do not know in advance when data is coming. The listening to an idle channel is useless and costly since typical radios consume much more energy in idle mode than in sleep mode (c.f. Table 2.1).

    - *Overhearing:* It describes a node receiving and decoding either a frame intended for other nodes or a redundant frame w.r.t. the transported information.

    - *Collisions:* they happen when a node is within the transmission range of two or more nodes and their transmissions conflicts so that the receiver

node can not decode any frame. Both the well-known *hidden-terminal scenario* and the inefficient coordination of transmitting activities among nodes may lead to collisions. These are energy wastes both at the destination and at the source node. Besides, they usually induce the retransmission of the problematic data or control frame.

- *Protocol overhead:* it describes the use of extra MAC control messages either for synchronising nodes, allocating communication resources or reducing the duty-cycle. Besides, this energy-consuming protocol overhead is even generated by the countermeasures to one of the energy inefficiencies listed above (i.e. coordination procedure for reducing collisions)

5. *Scheduling problem:* When applications demand predictable performance, a scheduled based MAC protocol is the obvious choice [3]. It relies on a scheduling strategy for allocating the communication resources among nodes. When Time Division Multiple Access (TDMA) schemes are employed, a *network schedule* indicates what each node can do (transmit, receive or sleep) in every *timeslot* and coordinates the transmission activities in the network. That allows to limit or eliminate collisions, idle listening and overhearing, i.e. different sources of energy waste. However, some form of time synchronisation among nodes is required to make sure that they agree on slot boundaries. This operation, as well as building the network schedule, involves the exchange of control messages and consumes energy. The scheduling problem is still a hot research topic in 6TiSCH networks. We detail on the scheduling strategies for such networks later in Section 2.4, after having described the TSCH protocol.

6. *Time synchronisation:* As explained above, time synchronisation is an important aspect for MAC protocols supporting scheduled access, such as TDMA. Note that it concerns energy efficiency, since time synchronisation allows the nodes to switch off their radio most of the time. The problem is that each device in a WSN has its clock (e.g. a crystal oscillator) and its notion of time (*local time*). Due to imperfections in the clock construction or environmental fluctuations, each clock beats at a slightly different frequency. Therefore, there is a difference comparing the local time of any two nodes (*clock offset*). A time synchronisation algorithm adjusts the time of one or both of these nodes, i.e. it keeps the nodes' clocks as tightly synchronised as possible. A relative offset of tens to hundreds of micro-seconds is acceptable in most WSN deployments [60]. To this end, two nodes exchange periodically (e.g. at intervals of

[3] Traditionally, (wireless) MAC protocols are divided into two categories: (i) contention-based (or random-based, unscheduled) and (ii) scheduled-based (or contention-free, reserved or frame-based) protocols.

tens of seconds) at least one synchronisation message containing a timestamp (*pair-wise synchronisation*). Repeating this process among node pairs leads to *network-wide time synchronisation*. Specifically, a spanning-tree with a *time master node* at its root is built. Every node synchronises to its parent (called *time source neighbour*) in a coordinated way, which prevents high synchronisation errors on deep nodes in the network [61]. It is also worth mentioning that a high network-wide timing accuracy allows to reduce idle-listening or to enable shorter slots with TDMA-based MAC protocols.

7. *Multichannel operation:* Most commercial radio devices support multichannel operation, that is, their transceiver provides several carrier frequencies (or channels) to choose from for transmissions or receptions. MAC protocols can use this feature for boosting network throughput (i.e. through parallel transmissions on orthogonal channels) [62] or for facing external interference and multipath fading [46]. However, multichannel operation adds complexity in designing the MAC protocol; indeed, neighbouring nodes must be on the same channel at the same time for exchanging messages. Hence, a multichannel MAC protocol requires a channel assignment method, which decides (i) the frequency of the assignment operations, (ii) the policy adopted for the selection of an adequate channel and (iii) the coordination between nodes at channel switching. Reference [63] provides a taxonomy for channel assignment methods in WSNs based on the three points mentioned above. In addition, note that the introduction of multiple channels complicates broadcasting, even locally. As a consequence, distributing the necessary control information for building and maintaining the network become more challenging.

2.3.3 Enabling Communication over Multiple Hops

Multi-hop communication is usually a necessary mechanism in WSNs to overcome the limited range of transmissions or to prevent using unaffordable transmission power. Thus, the communication between a sender and a receiver involves multiple intermediate nodes and relies on a routing protocol, which determines the adequate multi-hop path. However, well-established routing protocols designed for wired or wireless ad hoc networks are inadequate for (I)WSNs. Indeed, routing solutions for WSNs should be optimised for energy efficiency and adaptivity to topology dynamics (due to node failures, node mobility or link quality variations). In industrial WSNs, designing routing protocols also means finding high reliable and low latency paths that satisfy the specific application requirements. In what follows, we shortly

explain both the major design challenges and the suitable approaches for routing in (I)WSNs:

1. *Constrained devices:* A routing protocol is a crucial building block for achieving efficient and reliable data delivery in WSNs. Still, the routing process produces additional control traffic (e.g. for detecting neighbours, for transmitting topological information), requires memory resources (e.g. for storing routing and neighbour tables) and processing-time (e.g. for evaluating routing metrics or for selecting paths). Therefore, designing a routing protocol for (I)WSN consists of keeping its impact on node resources to a minimum.

2. *Unreliable and dynamic wireless links:* Recall from Section 2.3.1 that low power wireless links between sensor nodes are unreliable and time-variant. Besides, an imperfect MAC protocol can produce packet errors on links due to collisions or lack of coordination between transmitter and receiver. In addition, link failures caused by hardware problems, battery discharge, de-synchronisation events or node mobility are not rare. Therefore, routing protocols for (I)WSN should rely on mechanisms for estimating link quality as well as for promptly detecting link failures (see Section 3.3 for more details). However, these mechanisms must prevent (i) wasteful routing oscillations caused by the appearing and disappearing of links or (ii) overreactions to a relative modest link quality change [64].

3. *Broad application space:* The broad application space of (I)WSNs results in various network sizes, many traffic patterns and different service requirements. These three aspects considerably impact the design of a routing protocol:

    - Network size: With small network size (up to ca. 50 nodes) of several industrial scenarios, *centralised routing* with a single entity (i.e. a network manager) that computes and manages routes is still conceivable [65]. However, *distributed routing* algorithms are unavoidable with larger WSNs. Nevertheless, it is challenging to scale a routing protocol to large and dense networks due to the memory constraint of sensor device [66].

    - Traffic patterns: In a multi-hop IWSN, the application (or a specific protocol primitive) usually dictates one or more of the following traffic patterns: (i) *Multipoint-to-point (MP2P)*, where all the nodes send their packets to the root; (ii) *Point-to-multipoint (P2MP)*, where the root sends commands or control messages to all (or many) nodes in the network;

(iii) *Point-to-point (P2P)*, where data or signalling information are exchanged between two arbitrary nodes. The traffic pattern imposed by the application has a significant influence on the choice of the routing paradigm. Thus, P2P and MP2P necessitate *unicast* routing, but *multicast* routing is more appropriate for P2MP.

- Traffic characteristics: The choice between a *reactive* or *proactive* approach for route discovery (and maintenance) depends on the traffic characteristic of the application. On one side, applications with periodic, delay-sensitive traffic demand proactive routing protocols, which establish a connected graph at the beginning of network operations and then maintain it, regardless if its paths are actually needed or not. On the other side, reactive routing protocols, which discover routes on-demand, are for scenarios with sporadic, unpredictable and delay-tolerant traffic. It is worth mentioning that these two approaches have different costs in terms of resource overhead (i.e. memory and energy) and delay for data communication. Proactive routing significantly increases the resource consumption in each sensor node, which stores and transmits a large amount of routing information. Still, it can send data packets quicker than reactive routing, since no delay due to route discovery incurs before data transmissions. Moreover, if the application data encompasses also critical alarms, then the routing algorithm should introduce a prioritisation of this critical traffic, which reduces the queueing delay in each sensor node, and, thus, the end-to-end latency.

4. *Routing metrics:* The routing protocol computes paths according to a specific *Objective Function (OF)* for optimising certain key attributes of a network (e.g. maximising reliability or minimising latency) and, thus, for satisfying particular application requirements. In doing that, a routing algorithm for (I)WSNs evaluates not only link characteristics but also sensor node properties. Link throughput, link latency, link reliability level, node residual energy, hop count are some examples of links or node characteristics that may be used to define a routing metric for the path computation. Further, the combination of such primary link or node characteristics into a composite routing metrics can lead to the optimisation of more than one performance aspects [67], [68]. Routing protocols for (I)WSNs may use some link and node properties not only to compute metrics but also to define constraints for path calculation. Routes traversing nodes that have at least a specific residual energy or containing links that offer at least a prescribed reliability level are examples of possible constrained paths. In general, link and node properties change continuously

in (I)WSNs. Similar to that above stated about wireless links, it is essential to design a precise estimation mechanism of these routing metrics as well as an efficient route update process (see also Section 3.3).

5. *Gradient routing:* Many low-power wireless networks use gradient routing approach for data collection applications, as it is easy to implement on constrained hardware and robust against topological challenges [69]. GBR [70], CTP [71], Dozer [72] and RPL [73] are well-acknowledged representatives of this routing approach. In gradient routing, a scalar value (dubbed *rank*, height or cost) is associated with each node. The rank reflects the node's proximity to the sink, and, it monotonically increases along the paths from the sink to the source nodes. Thus, each node selects its available neighbour with the smallest rank both for forwarding packets and for updating its own rank. Neighbour discovery and link quality estimation are core elements for building and maintaining a stable and self-healing gradient. Note that gradient routing is particularly useful for MP2P, i.e. data collection. However, a supplementary routing paradigm is required to create downward paths for P2MP and P2P traffics.

6. *Multipath routing:* Multipath routing sets up multiple paths from source to sink nodes for coping with potential link or node failures. In traditional single-path routing, these failures induce data packet losses and the generation of extra control packets for discovering newer paths. Multipath routing avoids such pitfalls by exploiting path diversity. This strategy is recommended for IWSN [7], where coping with lossy link is even more challenging due to harsh and noisy industrial ennvironments. By adopting multipath routing, there is always an alternative path ready if a link or a node fails [74] or, alternatively, multiple copies of the same data packet traverse the network in parallel along different paths [75]. In addition, multipath routing may improve network performance in term of end-to-end delay and lifetime. Further, backup routes may serve for load balancing and congestion avoidance [76]. However, some issues come up by designing multipath routing protocols. In general, discovering and maintaining multiple paths require more computational complexity and resources (e.g. memory and energy) than for a single path. Besides, adjacent routes at the NET layer can interfere with each other at the MAC layer, if they are exploited concurrently (route coupling effect) [77]. Finally, transmitting multiple copies of data packets reduces network lifetime and capacity due to imposed overhead.

The discussion above legitimates the academic inclination to explore different strategies or a variety of objectives in designing routing protocols for (I)WSNs (the interested reader can refer to the comprehensive survey works [78]–[80]). However, the standardisation body IETF is promoting the adoption of RPL for WSNs. It is a gradient routing protocol that is able to meet a large set of requirements and offers adaptability for data collection. We detail RPL in Section 2.4.4, since it is the routing solution adopted in 6TiSCH for the IIoTs.

2.4 6TiSCH Protocol Stack for a Standardised Industrial IoT

2.4.1 The Role of Standards

It is widely recognised that standards are particularly important for IoT technologies [81]. From a technical perspective, standards ensure appropriate levels of interoperability between devices from different vendors. Further, adherence to standards helps to increase the performance and security of communication technologies. From a business perspective, standards are strategic market instruments. Indeed, users perceive standardised products as more stable and trustworthy. Further, standards can open access to international markets.

As highlighted in [82], there is nowadays a wide range of independent and competing standards for (industrial) IoT. In this context, the IEEE-SA [3] and IETF [14] are two of the most influential Standard Development Organisations (SDOs) regarding wireless technologies and networking for IoT. Concerning the technical focus, the IEEE-SA mostly targets the lower layers in the OSI protocol stack, namely the PHY and MAC layers. It is the IEEE 802.15 Working Group (WG) that focuses on low-power wireless technologies for WSNs and has been developing the IEEE 802.15.4 standard since 2003 [83]–[87]. The IEEE 802.15.4 is arguably the most adopted standard in WSNs. In contrast, the IETF targets the network, transport and application layers of the Internet. The TCP/IP protocol suite stands for the IETF specifications for computer communication. In the context of IoT, several IETF WGs are working on mechanisms for enabling IP connectivity on resource-constrained devices [88].

We think that the realisation of the IoT vision relies on the fruitful cooperation between these two prominent but different SDOs. On that front, the connection

between IEEE 802.15 [89] and the IETF is particularly relevant for the topics of this thesis. Firstly, the suite of IPv6 over Low power Wireless Personal Area Network (6LoWPAN) standards (i.e. [90], [91], [92]), published by IETF, has enabled IPv6 operations on IEEE 802.15.4-based WSNs and, therefore, the direct integration of sensor nodes into the Internet. More recently, the activities of the IETF 6TiSCH WG [93] are combining the above mentioned IETF solutions for IPv6 connectivity with TSCH, one of the MAC technique proposed by the IEEE-SA in the IEEE 802.15.4 standard for industrial low-power wireless communications.

In the following, we firstly overview the resulting *6TiSCH protocol stack*, which enables a open standardised integration of industrial sensor and data acquisition systems with the Internet, i.e. the realisation of the IIoT. Then, we detail on some of its components (i.e. TSCH, the routing protocol RPL and a selection of 6TiSCH standardisation activities) that are relevant for the research contributions presented in this thesis.

2.4.2 Overview of 6TiSCH Protocol Stack

Figure 2.2 depicts the primary building blocks that form the 6TiSCH protocol stack. At the upper layers, the Constrained Application Protocol (CoAP) [94] is used in conjunction with the User Datagram Protocol (UDP) [95] for allowing web-like interactions with constrained devices and their management. Object Security for Constrained RESTful Environments (OSCORE) [96] provides end-to-end and low-overhead security for CoAP messages.

At the network layer, IPv6 [97] and the adaptation layer 6LoWPAN [90] realise IP-based bi-directional communication to any devices in the IWSN. The RPL [73] is a gradient-based routing protocol that organises the mesh network in a multi-hop redundant topology, considering the constraints of low-power devices, following an objective function [98]. Routing informations are transported in Internet Control Message Protocol for IPv6 (ICMPv6) messages [73], [99].

The 6top Protocol (6P) [100] and other standardisation activities from 6TiSCH WG [93] (e.g. *minimal 6TiSCH* [101] and *Minimal Scheduling Function (MSF)* [102]) provide management and control primitives to harmonise the asynchronous IPv6-based upper layers with the synchronous IEEE 802.15.4 TSCH MAC protocol.

At the bottom, the 6TiSCH protocol stack adopts one of the several radio options defined by the last revision of IEEE 802.15.4 standard [87]. Usually, the transmissions take place in the 2.4 GHz band, where at most $127\,\mathrm{B}$ with a data rate of

250 kbps can be sent over a range in the order of 100 m–200 m outdoors (or 20 m–50 m indoors) [103]. In addition, the sub-GHz band can be exploited for application requiring a lower throughput (i.e. few kbps) and a more extended communication range (i.e. some kilometers).

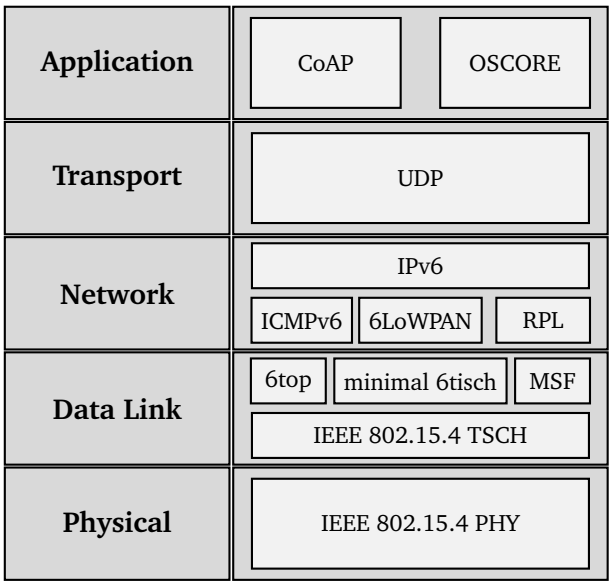


Fig. 2.2: 6TiSCH protocol stack

2.4.3 IEEE 802.15.4 TSCH

In its first versions, the IEEE 802.15.4 standard defines two basic MAC operational modes: a *beacon-enabled* (BE) and a *non-beacon enabled* (NBE) mode. In BE-mode, nodes shares active/sleep periods, maintain a certain level of synchronisation receiving *beacon frames* and compete for medium access using slotted CSMA/CA algorithm. In NBE-mode, nodes use a simpler unslotted CSMA/CA algorithm for channel access, but need to keep their radio always on for receiving data. However, several studies have identified the performance limitation and deficiencies of these both basic MAC modes of IEEE 802.15.4. They include the unbounded delay [104], the limited communication reliability [105], instabilities due to interference or multi-path fading [69] and the necessity of powered relay nodes in peer-to-peer topologies [106]. In sum, the classic MAC layer in the IEEE 802.15.4 was precluding the use of this standard in industrial applications at the end of the 2000s.

To overcome that, in 2012, the amendment IEEE 802.15.4e introduced five new MAC modes. Among them, TSCH was integrated into the successive revisions of the standard [86], [87] and has gained increasing attention in academia and industry.

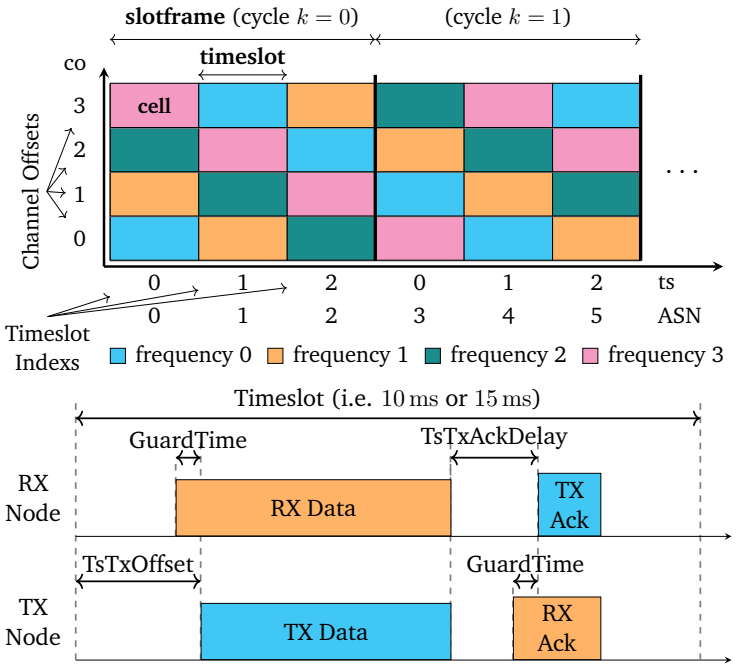


Fig. 2.3: TSCH Taxonomy (above) and Timeslot Timing (below)

As MAC protocol, TSCH has a design inherited from the Time Synchronised Mesh Protocol (TSMP) [107], WirelessHART [108] and ISA100 Wireless [5], which are proven technology in guaranteeing industrial performance requirements with low-power wireless networking.

In the following, we describe the key concepts of TSCH.

TSCH slotframe structure

TSCH employs both a *time-slotted channel access* mode and a *channel-hopping* mechanism. As depicted in Figure 2.3, time is divided into intervals of fixed length, called timeslots. A *timeslot* is typically $10\,\mathrm{ms}$ long and ensures the transmission of a data frame as well as the reception of the related acknowledgement. Timeslots are grouped into a *slotframe*. Globally, an Absolute Slot Number (ASN) labels each timeslot over consecutive slotframes. Specifically, the $ASN = k \cdot L_{SF} + ts_i$ counts the total number of timeslots elapsed since the start of the network, where k defines the sloframe cycle, L_{SF} is the length of the slotframe, and ts_i is an index that maps the relative position of a timeslot in the current slotframe. The ASN is a value shared by all nodes in the TSCH-network.

A schedule defines how a node operates in each timeslot within a slotframe: transmit, receive or turn the radio off for saving energy. Besides, it indicates which of the up to

$N_C \leq 16$ different physical frequencies or channels a node has to use in each timeslot for transmitting or receiving. As a result, TSCH provides a matrix-like schedule (c.f. Figures 2.3 and 2.4), with a height equal to the number of available frequencies and a width that is the length of the slotframe (in timeslots). An element in the schedule matrix is called *cell* or *link*; it is identified by a pair $[ts, co]$, i.e. the timeslot index ts in the slotframe and the channel offset co for computing the frequency the radio transceiver should use in that cell. Specifically, the channel offset translates into the actual frequency a node has to use as follows:

$$f = F[(ASN + co) \mod N_c]. \tag{2.1}$$

The function F maps an integer value into a physical IEEE 802.15.4 frequency using a lookup table. Further, Eq. 2.1 implements the channel hopping mechanism of TSCH, since it gives a different physical frequency for the same cell in successive slotframes, following a pseudo-random hopping sequence. Thus, TSCH tackles the effect of multipath fading and external interference. Indeed, when a transmission fails on a specific frequency, the next attempt will occur on a different one in the successive slotframe, probably with other conditions. As explained in [109], selecting the length of the slotframe L_{SF} coprime to the number of available channels N_c enables to use the full diversity of the radio spectrum, i.e. to improve the network reliability.

Moreover, the multichannel operation of TSCH increases network capacity, since simultaneous communications can take place without interfering in the same timeslot by using different channel offsets.

TSCH defines both *dedicated* and *shared* cells. Dedicated cells are allocated to a single sender-receiver couple and are contention-free. On the other hand, shared cells can be accessed by multiple nodes at the same time and require the execution of a CSMA/CA like protocol.

Let us consider an exemplary multi-hop network and a plausible matrix-like TSCH schedule for it. Each node translates this network-wide matrix-like schedule into a local one, where the scheduled activities (transmit, receive or sleep) repeat over time, as shown in Figure 2.4 for the sink and node B. Further, multiple slotframes with different durations can coexist in a node schedule. When a node has conflicting activities scheduled for a specific timeslot, it uses the priority associated with each slotframe to resolve this conflict.

The TSCH MAC mode of the IEEE 802.15.4 standard defines how a node executes a given TSCH schedule, but not, how to build such a schedule. Constructing efficient

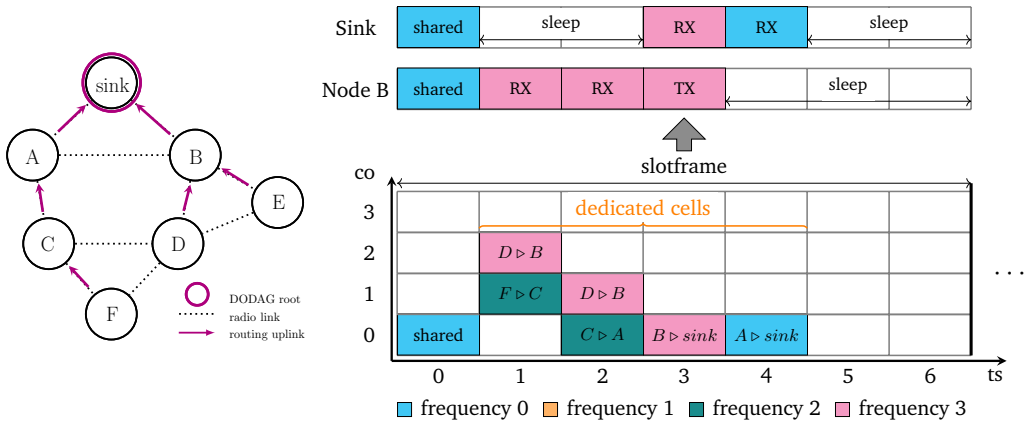


Fig. 2.4: A multi-hop 6TiSCH network (left) and a possible TSCH schedule (right)

schedules for applications using TSCH networks is still an open research problem, as we explain later in Section 2.4.5.

Synchronisation and timeslot timing

TSCH operates in a time-slotted mode and requires, therefore, a network-wide node synchronisation. A node is synchronised when it learns the current ASN value, and it can align the edges of its slotframe structure to that of the network. In a TSCH-network, there is a time master node, to which all the other node synchronise over a spanning tree, i.e. each node in the network identifies a time parent node, which acts as its time reference.

In TSCH, a slight de-synchronisation between a sender and a receiver node is allowed. Precisely, TSCH incorporates a `GuardTime` in the process executed for exchanging a data packet during a timeslot. As we see in Figure 2.3, the sender node transmits its data packet exactly `TsTxOffset` μs after the beginning of the timeslot; in contrast, the receiver node starts listening to the channel `GuardTime` μs before and waits for the PHY-layer preamble of the packet until `GuardTime` μs after the expected `TsTxOffset`. Therefore, a packet exchange is viable, when the de-synchronisation between the sender and receiver node is less than the guard time.

Further, a packet exchange with a time parent lets a node re-synchronise to it (i.e. pair-wise synchronisation). Since the value of `TsTxOffset` is fixed and known a priori by all the nodes in the network, a node can compute the clock difference by timestamping the actual time of reception and then retroactively align its slot boundaries to that of its time parent.

As already explained in Section 2.3.2, pair-wise synchronisations must happen regularly in a WSN. To this end, a node has to send short *Keep-Alive (KA)* packets, if it does not receive any message from its time parent within a t_{ka} period

Time synchronisation is a prerequisite for the TDMA mechanism in TSCH, but it does not require any complex time synchronisation protocol. Indeed, regular packet exchanges are sufficient for this task in TSCH. Moreover, as a positive side effect, the synchronisation at the MAC layer minimises the time during which transceivers are on. For this reason, TSCH provides an ultra-low level of power consumption, if compared to other asynchronous MAC protocols. Specifically, in TSCH, the sender nodes know precisely when to transmit, i.e. `TsTxOffset` μs after the beginning of the timeslot. Further, nodes can switch their radios off to save energy after the transmission of packets, and wait for the next scheduled event. Finally, idle-listening occurs at receivers only during the `GuardTime` of active cells. The authors of [110] model the energy consumption of TSCH and the impact on it of the one-hop synchronisation. Several works [61], [111], [112] demonstrate that improving the synchronisation process allows for even lower duty cycle and longer network lifetime in TSCH.

In our proposal for improving the network formation in TSCH presented in Chapter 4, we limit us to enhance the scheduling for delivering control packets (e.g. Enhanced Beacon (EB) frames), whose reception is essential for acquiring or maintaining the synchronisation. Indirectly, our proposal reduces the energy-overhead caused by one-hop TSCH synchronisation and its KA messages; indeed, since control packets from the time parent nodes are less prone to collisions, their reception during a t_{ka} period is more likely, as well as a reduction of the number of transmitted KA messages.

TSCH Frame Format:

The TSCH protocol adopts the general MAC frame format prescribed by the IEEE 802.15.4, as it is presented in Figure 2.5, for fragmenting and formatting the payload received by the upper layer into MAC frames. For brevity, we do not explain the purpose of each field in the MAC header (or trailer). Rather, we put the focus only on the variable field Information Element (IE) due to its relevance in TSCH and in our proposals presented in the next chapters. An IE is a flexible and extensible container of information, located between the end of the MAC header and the frame payload. Any IE field contains *length*, *type* and *content* fields, as depicted in Figure 2.5. It was defined with the revision of the standard IEEE 802.15.4 in 2015 to exchange

information between one-hop neighbours or between far devices. For instance, a beacon broadcast by a Personal Area Network (PAN) coordinator turns into an EB in a TSCH network, when it encapsulates, within multiple concatenated IEs, additional information, e.g. regarding the TSCH synchronisation, slotframe structure, channel hopping pattern and schedule. Nevertheless, the standard does not limit the use of IEs to beacon frame, but also data, acknowledgement and command frames may contain them. They are either part of the MAC header (Header IEs) or of the payload (Payload IEs). The IEEE specifies many different types (and subtypes) of IEs for TSCH and the other MAC modes in 802.15.4 [86]. Still, a research-related or vendor-specific implementation of IEs for other purposes is possible. For example, some works show their use for transporting CoAP messages [113] or for inserting telemetry data [114].

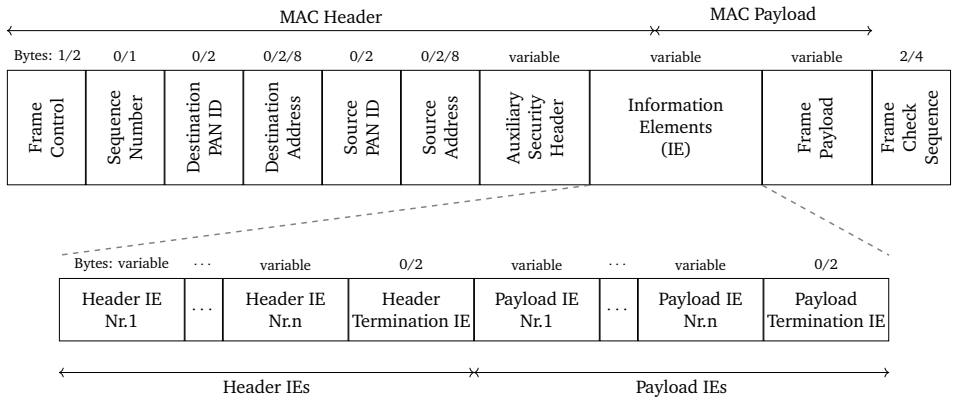


Fig. 2.5: General format of an IEEE 802.15.4 MAC frame (above) and of IE field (below)

2.4.4 Routing in 6TiSCH-Stack: RPL

The IETF ROLL working group specified in 2011 the RPL [73]. RPL has been designed to connect up to thousands constrained devices in a network where the communication is prone to high error rate. Therefore, the objectives of RPL are to minimise the memory requirements of the implementation, to reduce the complexity of the routing functionalities and to limit the signalling overhead. Besides, RPL includes mechanisms for quickly reconfiguring the routing topology or detecting loops to deal with the unreliability (and volatility) of wireless links (c.f. Section 2.3.3).

RPL targets a wide range of application scenarios, notably, home and building automation as well as industrial and urban settings [7], [115]–[117]. Thus, RPL supports the traffic patterns presented Section 2.3.3. Nevertheless, the protocol design assumes that the majority of the application traffic is MP2P. On the contrary,

P2MP is assumed to be sporadic, and P2P traffic is considered rare. For this reason, the IETF has adopted the gradient routing concept for designing RPL. In addition, RPL offers two modes of operation for supporting the optional P2MP or P2P communications: *storing* and *non-storing*. In storing mode, nodes maintain routing tables locally with information about their sub-tree. In the non-storing mode, instead, only the root collects topology information for downward routes.

RPL-Topology

As above stated, RPL is a gradient routing protocol. RPL builds and maintains a routing topology in the form of a *Destination Oriented Directed Acyclic Graph (DODAG)* for upstream data delivery. The DODAG is a directed tree, oriented toward a root node, the DODAG root.

The DODAG root (or sink) is a central and less constrained node, mainly responsible for gathering data. It may also act as a border router for the DODAG, i.e. connecting it to external networks or the Internet.

Each other DODAG node maintains a set of parents among neighbours and selects one of them as its preferred parent for forwarding packets towards the DODAG root. To this end, RPL assigns to each node a rank, a scalar measure that encodes the distance of the node from the DODAG root. The rank strictly increases downwards, i.e. the multiple parents of a node are neighbours with a lower rank.

Objective Function and Metrics

An OF defines the metrics and the constraints for computing a node's rank and, therefore, the optimisation objectives for the DODAG construction. Thus, an OF is a concept that allows adapting RPL to the wide variety of use cases it serves. The IETF has proposed two exemplary OFs: the default Objective Function 0 (OF0) and the Minimum Rank with Hysteresis Objective Function (MRHOF). The OF0 [98] is a simple OF, where a node chooses the preferred parent as the one advertising the lowest rank; then it computes its rank by adding a strictly positive scalar value to the one of its preferred parent. The MRHOF [118] employs hysteresis for stable parent selection. That is, a node changes its preferred parent only when this switching leads to a significantly smaller path cost than the previous one. A threshold quantifies this difference and sets a compromise between route stability and optimality. Both OFs do not specify which metrics and constraints should be involved in the calculation of the rank increase. Section 3.3.2 describes different routing metrics that have been

defined in literature for RPL and highlights the relevance of Expected Transmission Count (ETX) [119].

RPL control messages

RPL uses ICMPv6 for exchanging the signalling information. Namely, the following RPL control messages are specified:

- *DODAG Information Object (DIO)*: this message is similar to an IPv6 Router Advertisement and announces the DODAG-ID, the sender's rank and other configuration parameter used for DODAG construction and maintenance.
- *DODAG Information Solicitation (DIS)*: it is either a broadcast or a unicast message, used to trigger the transmission of DIO messages from neighbours.
- *Destination Advertisement Object (DAO)*: this message serves for establishing downward routes. A node transmits a DAO as unicast packet either to the DODAG root (non-storing mode) or to selected parents (storing mode) for announcing itself as a possible destination. A DAO is propagated upward along the DODAG and may contain the address of the destination's parent (in addition to the advertised destination prefix).
- *DAO Acknowledgment (DAO-ACK)*: a DAO recipient (i.e. a DAO parent or the DODAG root) may reply to a DAO sender with this message to confirm a route installation or removal.

We explain the DODAG formation procedure and how these RPL control messages are involved in Section 4.2.

2.4.5 6TiSCH Control Plane

As explained in Section 2.4.3, the IEEE 802.15.4 standard simply clarifies how nodes execute a given TSCH schedule, and not the link allocation mechanism, that is, how to build a TSCH network-wide schedule. Some standardisation activities of the 6TiSCH WG aim to define the management and control primitives for building and maintaining multi-hop schedules for RPL-organised, TSCH-based networks. To this end, two separate elements are utilised: the 6P protocol and the Scheduling Function (SF).

The **6P** [100] specifies a set of procedures that two neighbours nodes have to use for negotiating dedicated cells in the TSCH-matrix. The protocol defines messages to ADD, DELETE, RELOCATE, COUNT, LIST, SIGNAL or CLEAR cells in the neighbour's schedule. The message sequence chart for the negotiation coincides to either a two-way- or a three-way-handshake, as depicted in Figure 2.6. 6P uses an IE container, which IEEE has reserved for the IETF, to encapsulate 6P messages in MAC-layer packets. Finally, 6P adopts sequence numbers and a set of timeouts to detect possible schedule inconsistencies that may happen due to the lossy nature of links.

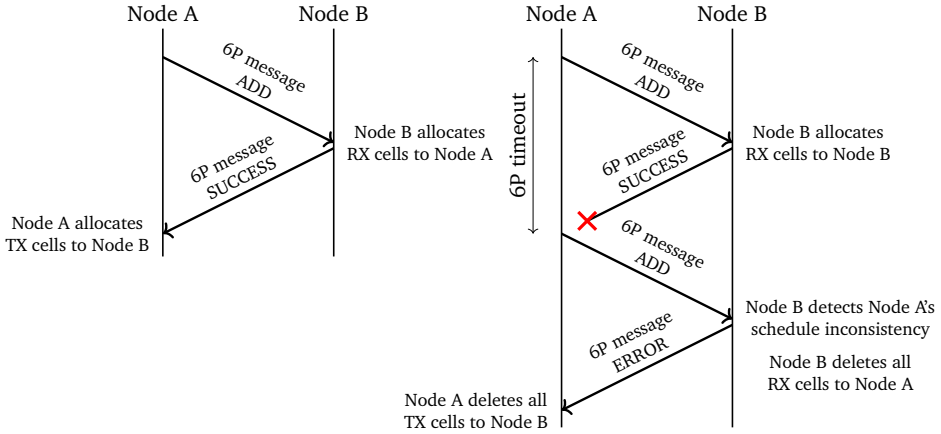


Fig. 2.6: Examples of 2-step 6P transactions: successful (left) or failed (right)

The **SF** is the algorithm that decides how many cells two neighbours node allocate, and when a 6P negotiation between nodes has to be triggered. 6TiSCH WG is currently defining the **MSF** [102]. MSF builds and maintains a 6TiSCH schedule in an autonomous and distributed manner. Through MSF, nodes allocate two types of cells for unicast communication with their neighbours: autonomous and negotiated cells. Autonomous cells provide a minimal bandwidth for exchanging control traffic such as joining or 6P messages. Their coordinates (ts_i, co_j) are a function of the MAC address, and nodes compute them autonomously. Instead, the negotiated cells are allocated between neighbours through 6P negotiations when needed, or removed when they are not used any more. To this end, a node considers both its traffic requirements and the current utilisation of the scheduled cells with its preferred parent.

Beyond 6TiSCH WG, numerous SFs or scheduling approaches for TSCH have already been proposed in the literature. A detailed description of these proposals is outside of the scope of this Chapter. We refer the reader to [120]–[122] to this end. In relation to the contributions in this thesis, we highlight that most of the scheduling solutions proposed for 6TiSCH move from the following assumptions: (i) the 6TiSCH network is already operational, i.e. all the node are network-wide synchronised and in a

DODAG, (ii) the nodes known their neighbours and the link quality to them, (iii) the signalling information for obtaining network characteristics or for negotiating the schedule does not experience collisions. With our proposals in Chapter 4 and 5, we aim to improve the 6TiSCH network formation and to ensure an accurate link quality estimation already at bootstrap, respectively. Further, we reduce the probability of collision of control frames during the initial stage of a 6TiSCH network. Thus, our work has a positive impact on the performance of the SFs. Nevertheless, we adopt autonomous and distributed scheduling mechanisms to bootstrap the 6TiSCH network and coordinate the transmission of signalling information.

Besides 6P and SFs, the 6TiSCH WG provides a set of specification for bootstrapping a 6TiSCH network in a secure way. The **6TiSCH Minimal Configuration (6TiSCH-MC)** [101] defines a minimal mode of operation and ensures essential interoperability between nodes by specifying a particular configuration of the numerous parameters of the IEEE 802.15.4 standard, such as timeslot duration, slotframe length, etc. We detail the 6TiSCH-MC in Section 4.2, where we discuss our extensive simulation study on 6TiSCH-MC and strategies for improving the network formation phase. Further, a secure join process is currently under standardisation [123]. By this process, a new node (called *pledge*) requests the admission into the network and sets up keys used to authenticate and encrypt subsequent transmissions. Since the topic security for IWSN is out of scope in this dissertation, we refer the interested reader to [124]–[126].

2.5 Other Standardisation Efforts for Industrial IoT

Despite the catalyst role the 6TiSCH protocol stack is playing in the academic community [127], there are other low-power wireless standards that can compete with it for the realisation of the (Industrial) IoT vision. In the following, we briefly describe these alternative solutions. This allows a rough comparison with 6TiSCH and a discussion about their relations with the contributions of our research.

2.5.1 Other IEEE 802 Standards for Wireless Sensor Networks

Deterministic and Synchronous Multi-channel Extension (DSME) is one of the MAC modes that the amendment IEEE 802.15.4e introduced in 2012. As well as TSCH, DSME is intended for industrial automation and process control applications. It extends the superframe structure, which is already defined by the classic BE mode

in IEEE 802.15.4, to multi-hop and multi-channels networks. Besides, it defines a distributed procedure for scheduling a collision-free communications. Similarly to TSCH, DSME exploits time-slotted channel access and frequency diversity. Despite adopting similar basic principles, the current research and commercial interest behind DSME is not comparable with that behind TSCH. DSME-related works in literature are authored by only a handful of research groups [128]–[130], and merely a software implementation is available [131] at the time of writing. Rather than a technical superiority, these practical considerations explain our preference for TSCH over DSME as the reference MAC protocol in our research works.

Bluetooth Low Energy (BLE) was introduced with Bluetooth 4.0 in 2011 for significantly reducing the size, the complexity and the power consumption of the classical Bluetooth communication stack [132]. Nowadays, the high availability of devices supporting BLE is not limited to consumer electronics devices but also battery-operated sensors and actuators for industrial application are on the market (e.g. based on SoCs such as TI CC2650 [133], nRF52840 [134] and AT86RF215 [135]). BLE operates in the 2.4 GHz ISM band and employs frequency hopping over 40 channels, each $2\,\mathrm{MHz}$ wide. It can provide a bitrate of $1\,\mathrm{Mbps}$ transmitting data packets with a size of up to $255\,\mathrm{B}$. In literature, several works compare the performance of BLE and IEEE 802.15.4. Although some measurements have shown a lower energy consumption of BLE [136], [137], there is no clear winner when the TSCH mode of IEEE 802.15.4 is adopted, especially on a single link [138]. Further, BLE requires particular extensions or configurations of its TDMA-based polling mechanism for increasing its timeliness as required in industrial environments [139]–[142]. Note that BLE supports only star networks with a central node (master) that coordinates the data transfer with one or more peripheral devices (slaves). This design choice minimises the complexity and the size of the protocol stack; however,it is a relevant drawback if compared to the network topologies supported by IEEE 802.15.4. For this reason, two different mesh networking standards, namely Bluetooth Mesh [143] and 6BLEMesh [144], were introduced to expand the communication coverage of a BLE network. Reference [145] offers a comparison of the features and performances of both solutions. It is noteworthy that 6BLEMesh is currently developed by IETF 6lo WG [146] with a similar approach followed by 6TiSCH WG. However, the maturity of 6BLEmesh is limited when compared to 6TiSCH.

The standard **IEEE 802.11ah** [147] (branded as Wi-Fi HaLow) was published in 2017 to adopt Wi-Fi to the needs of the IoTs. Compared to the other specifications in the IEEE 802.11 family of standards (i.e. 11n and 11ac), the IEEE 802.11ah improves the low power operation, the scalability and the communication range. As for BLE, IEEE 802.11ah supports star network topologies. Due to the operations in the sub-

GHz band, an access point (AP) can provide connectivity between heterogeneous stations (STAs) over a range up to 1 km. The provided data rate ranges from 0.3 Mbps to 7.80 Mbps (depending on the modulation and encoding mechanism adopted). At MAC layer, IEEE 802.11ah augments the legacy CSMA/CA mechanism of IEEE 802.11 with some energy-saving tools (e.g. implicit acknowledgements or a traffic indication map in beacons) and an optional restricted access window mechanism for grouping clients in dense networks. However, the availability of hardware supporting IEEE 802.11ah is very limited at the time of writing. Thus, only analytical or simulative studies on the performance of IEEE 802.11ah are provided in literature [148]–[150].

2.5.2 Standards for Low-Power Wide-Area Networks

Recently, different wireless technologies in the context of both Low Power Wide Area Network (LPWAN) and cellular systems were proposed for IoT connectivity. On the one hand, the 3GPP standardisation body [151] has introduced with **Narrow-Band IoT (NB-IoT)** [152] and **Massive Machine-Type Communications (mMTC)** [153] two relevant services profiles, respectively in 4G and 5G cellular systems, for machine-to-machine communications in the licensed radio spectrum. On the other hand, the LoRa Alliance [154] defines and maintains the **Long Range Wide Area Network (LoRaWAN) specifications** [155], i.e. the protocol stack above the patented PHY layer LoRa [156], which operates in unlicensed bands in the sub-GHz spectrum. All these solutions are based on long-range radio links (on the order of tens of kilometres), which are possible thanks to the use of the sub-GHz band for transmission and very high receiver sensitivity. Further, such solutions address applications that rely, in most cases, on sporadic transmissions of short packets. At the same time, their network architecture corresponds to a star topology, with a high number of battery-operated devices directly connected to a concentrator, which acts as the gateway to the Internet. For these reasons, the design of the lower communication layers of LPWAN standards differs significantly from that of 6TiSCH (c.f. Section 2.4). Consequently, the addressed challenges in our research works, which deal with the formation and maintenance of multi-hop mesh networks, are irrelevant in LPWANs.

2.5.3 Industrial Standards: WirelessHART, ISA100 Wireless and WIA-PA

For replacing wired solutions in industrial applications, low-power wireless standards must guarantee deterministic delay, high end-to-end reliability, management-free operation over several years and certified security. In the last decade, standards like WirelessHART [157], ISA100.11a [158] and WIA-PA [159] have established themselves as adequate wireless solutions in process automation applications. **WirelessHART** was released by the HART communication foundation [108] as part of the HART7 revision in 2007, and it became an International Electrotechnical Commission (IEC) standard (IEC 62591) in 2010. Almost in parallel, the IEC approved the **ISA100 Wireless** and the **WIA-PA** as international standards (IEC62734 and IEC 62601), after their initial success, respectively, in the US and Chinese markets. All three standards were specifically developed for industrial applications. They specify not only the wireless communication at the lower OSI layers (PHY, MAC and NET) but also the transport and the application layers. We refer the reader to [160], [161] for a comprehensive description and comparison of these three IWSN standards.

These three standards reveal that wireless low-power communications are nowadays able to meet industrial requirements in process automation when both time slotted access and channel hopping are adopted. Further, ISA100 Wireless is consistent with the vision of the industrial Internet, since it supports IPv6 and the integration of field devices into the Internet. However, there are technological incompatibilities (e.g. time synchronisation, slot time, meshing methods) that prevent the interoperability between these three standards or the interconnection of existing networks. Motivated by this consideration, the IETF is promoting with 6TiSCH an open, standardised protocol stack for IIoT. This is the main reason why our research works focus on the 6TiSCH protocol stack and its challenges; still, our contributions can be easily applied to industrial networks that adopt WirelessHART or ISA100 Wireless.

2.6 Chapter Summary

This chapter gave a necessary introduction about low-power wireless networking for IIoT. We started with an overview of the type of hardware an (industrial) WSN comprises, i.e. microcontroller-based boards with a battery as the power source, a sensing unit to interface with the physical world and a radio chip for communicating with other devices. We described the role of an embedded OS, which has to use the constrained hardware resources efficiently and ease the development and portability

of IoT applications. Next, we explained why low-power wireless links are more unreliable and time-variant than other traditional wireless channels (e.g. cellular or WLAN ones). Later, we discussed the design challenges and the suitable approaches both at the MAC and NET layers for making a WSN dependable. Space limitations precluded us from providing a comprehensive description of existing low-power wireless standards and technologies. Instead, we restricted ourselves to underline the role of standards and the relevance of the cooperation between IEEE-SA and IETF to realise the IoT vision.

In this regard, we introduced the 6TiSCH protocol stack and its key elements, i.e. the MAC technology TSCH, the routing protocol RPL and the management plane specified through the 6TiSCH standardisation activities. The 6TiSCH protocol stack is already well acknowledged in industry and academia. This set of open-source standards achieves IPv6-based end-to-end connectivity between devices from different vendors and enables dependable low-power wireless communication for industrial applications. It looks like the appropriate solution for the realisation of the IIoT vision and, therefore, it lays the basis of our research work.

The following Chapter 3 positions this thesis within the low-power wireless networking and 6TiSCH community, highlighting the difference to the major solutions proposed so far.

3 Related Work

3.1 Chapter Overview

This chapter presents the state of the art of the two main topics that we handle in our research: bootstrapping low-power multi-hop wireless sensor networks and Link Quality Estimation (LQE) algorithms for such networks. In this context, an entity (dubbed as *coordinator*, *sink* or *root* node) is the initiator of several control plane primitives that characterise the early stage of the network lifetime: *network advertising*, *joining*, *synchronisation*, *neighbour discovery*, *topology establishment*. In general, these operations are executed both at the Medium Access Control (MAC) and Network (NET) layers (recall Sections 2.4.3 and 2.4.4). Further, they have a relevant impact on network performance in its operational stage. At the same time, LQE is of prime interest for establishing a logical topology. On the one hand, it helps MAC protocols to choose which neighbours a node should communicate with. On the other hand, routing protocols rely on the link quality information to construct multi-hop-paths that guarantee high reliability and energy-efficiency in the network. Moreover, when Time Division Multiple Access (TDMA)-based MAC protocols (as Time Slotted Channel Hopping (TSCH)) are adopted, link quality information is an essential input for building the communication schedule.

Starting from these premises, it is therefore not surprising that the list of papers published during the last decade on these two topics is vast. For the sake of brevity and clarity, we restrict here our attention only to the works which either (i) explicitly address protocols in the 6TiSCH stack or (ii) have had an extensive influence on the proposals we present in the next chapters.

The rest of the chapter is organised as follows. It first presents selected works on the initial stage of a multi-hop Wireless Sensor Networks (WSNs) in Section 3.2. Subsequently, Section 3.3 gives an overview of the different methods and strategies proposed in the literature for an accurate LQE in WSNs.

3.2 Related Work on 6TiSCH Network Formation

In this section, we divide the several works concerning the initial phase of a multi-hop WSNs in three different groups: (i) beacon scheduling for 802.15.4-based multi-hop networks, (ii) network formation with the classical Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) and Routing Protocol for Low Power and Lossy Networks (RPL), (iii) network formation in 6TiSCH. The first group contains the published research focussing on the beacon collision problem, which hinders the synchronisation in IEEE 802.15.4-based networks. In the second group, we review the existing studies on the RPL Destination Oriented Directed Acyclic Graph (DODAG) construction process in networks that adopt the classical CSMA/CA at MAC layer. Finally, the third group reviews the small body of literature that has investigated the 6TiSCH network formation process and the interactions of the different protocols in the stack during this process.

3.2.1 Beacon Scheduling for 802.15.4-based Multi-Hop Networks

Coordinating beacon transmissions in a multi-hop topology to avoid collisions is a key concern both for the beacon-enabled mode and TSCH mode of the IEEE 802.15.4 MAC standards. Consequently, a considerable amount of literature has addressed this problem considering a cluster-tree topology that works with the beacon-enabled mode and comprises multiple coordinators. To synchronise the transmission of beacons over multiple clusters, some works [162], [163] have proposed centralised mechanisms that are based either on a pure time-division approach or on a multi-channel operation. However, these works cannot be straightly adopted in 6TiSCH networks, since they do not take into account two key elements of TSCH, i.e. the channel hopping scheme and the support of mesh topologies. On the one hand, the channel hopping makes the initial synchronisation more complicated as any joining node has to select the right channel at the right time, i.e. the frequency used in a specific timeslot by an advertiser node in its neighbourhood. On the other hand, all joining nodes may become advertiser nodes after having terminated the initial configuration in a TSCH-based mesh topology. Thus, distributed beacon scheduling strategies have been proposed for such topologies. In these solutions, nodes have to collect information about their two-hops coordinators. To this end, the authors of [164], [165] defined a new association procedure where each joining node first scans the neighbourhood and then asks the neighbours' beacon schedule explicitly. Recently, the standard IEEE 802.15.4 [86], [87] addressed the beacon collision

problem in mesh networks, but only for the Deterministic and Synchronous Multichannel Extension (DSME) mode (c.f. Section 2.5.1). The solution suggested for the DSME mode consists in (i) defining a bitmap sequence, which represents the list of allocated or unused beacon slots in the neighbourhood of a node, (ii) piggybacking this bitmap in each beacon and (iii) prescribing message sequence charts for notifying the use of a vacant slot or for resolving possible collisions. However, previous works like [166]–[168] have highlighted the incompleteness of this proposal and suggested some enhancements. The approach proposed in [166] utilises an indicator instead of a bitmap, and it rests on active associations instead of passive scans of neighbouring beacons. The authors of [167] revised the DSME beacon scheduling algorithm by restructuring the superframe structure and by changing the explicit messages for association and feedback. Cano et al. [168] proposed to use a learning approach for collision-free operation of beacon transmission in mesh networks. With this approach, a node periodically decides which slot to use for sending a beacon considering the feedback provided by its neighbours about its previous transmissions. Note that this work is the closest to our proposal described in Chapter 4, where we also suggest coordinating the initial allocation of broadcast cells with feedback information included in the Enhanced Beacon (EB) messages (recall Section 2.4.3). However, our solution does not stress the convergence to collision-free operation as does [168]. The rationale behind our different design choice is that the 6TiSCH network formation process can cope well with sporadic EB collisions, as explained later.

Considering the TSCH mode, the standard IEEE 802.15.4 leaves unspecified the strategy that controls the beacon transmission, which strongly influences the time and energy spent by devices on the TSCH network formation phase [169]. Nevertheless, different beacon scheduling algorithms have been proposed in the literature so far. There are some advertising policies based on a dynamic and configurable beacon generation/transmission rate. For example, Duy et al. [170] propose a fuzzy logic mechanism that each advertiser node uses for adjusting its beacon transmission rate based on the network density. As an other solution, Kalita et al. [171] suggest to adjust the beacon interval according to the observed congestion status in shared slots. The Bell-X algorithm [172] controls the beacon period using a tunable periodic function that resembles a stepped bell and enables different beacon transmission zones in the network. The Contiki implementation of TSCH [173] suggests the adoption of a logic similar to [172]; when RPL as routing protocol is adopted, a node may map its beacon period on the current RPL Trickle timer, still bounding it to a TSCH specific maximal value for avoiding de-synchronisation events. Our solution does not follow this direction (i.e. we adopt a fixed period for the beacon

generation) but it could smoothly incorporate this logic for reducing the amount of EB frames at the steady state of the network.

In constrast, other proposals have focused on reserving a given set of cells per slotframe for EB transmissions. Where these N_{adv} advertising cells are located and how each advertiser node selects these transmission opportunities depends on the specific algorithm. In [174], multiple consecutive slotframe are grouped to form a multi-slotframe, and the advertising cells are located either in the first timeslot of a multi-superframe (using different channel offsets) or in the first timeslot of every slotframe in the multi-superframe (but only using the single channel offset $co = 0$). In [175], the Personal Area Network (PAN) coordinator initially determines an EB schedule by solving an optimisation problem that minimises the average joining time of nodes. The EB schedule is a set of advertising cells that are located equally spaced within the slotframe. The coordinates of these advertising slots (i.e. their time and channel offset) are included inside every EB broadcast in the network, so that every node may autonomously choose one of these links (i.e. for transmitting one EB per slotframe after joining). In both works [174] and [175], the PAN coordinator determines the fixed set of advertising cells, and any new synchronised node randomly chooses a cell from this set. Therefore, these two approaches do not handle the problem of EB collisions. In contrast, the algorithm proposed in [176] ensures that beacon transmissions regularly take place on all frequencies used by the TSCH network without collisions. To this end, each node calculates the coordinates of its unique advertising cell during the association procedure. The authors of [177] aim to achieve a collision-free transmission of EBs, too. They obtain this by reserving a number of advertisement cells at least equal to the number of nodes in the network and using the node's identifier to assign each of these cells to at most one node. However, these last two approaches are not compliant with 6TiSCH Minimal Configuration (6TiSCH-MC) [101] (see Section 4.2), impose a relative high duty-cycle, and require additional management overhead. Further, solving the EB collision problem, these proposals may hinder a reliable neighbour discovery. Indeed, various neighbouring advertiser nodes may select the same timeslot (but different channel offsets) for their transmission, and, as half-duplex devices, they can not learn about each other. In our proposal for improving 6TiSCH network formation (see Chapter 4), we also reserve a set of TSCH links for control messages, and, in doing that, we allocate them equally-spaced within the slotframe, as recommended in [175]. However, we introduce an algorithm to dynamically set the number of allocated cells depending on the node density in the neighbourhood. Further, we drive the advertisement cells' reservation for reducing the probability of EB collisions in dense networks. Finally, we optimise the propagation of the cell list

that, otherwise, can lead to the problematic fragmentation of EBs, when the value N_{adv} is relative high.

To complete our literature review on beacon advertising in TSCH, we need to mention the approaches that restrict the channels for the scanning/advertising process in environments with excessive interference. For example, in [178], the joining node does not randomly select the channel for passive scanning; instead, it periodically sets the best quality channel performing measurements on all channels at the end of each slotframe. The authors assume that the same channel is chosen by the advertiser nodes with high probability since they measures the channel quality in the same way. However, we consider this authors' assumption as questionable (because of sorting channel with very similar quality using low-end devices); besides, the rationale behinds this algorithm does not fit the channel-hopping mechanism of TSCH.

3.2.2 Network Formation with the Classical IEEE 802.15.4 and RPL

Several works have studied the RPL DODAG construction process in networks that adopt the classical CSMA/CA as MAC layer. For instance, the authors of [179] evaluate the control overhead of RPL using the Cooja simulator. Results show a transient phase for the network bootstrap of about 10 minutes, considering the time required to let significantly drop the overhead. That work considers fixed values of the Trickle timer parameters, random topologies composed of 20 or 100 nodes and a simple radio model. A more comprehensive insight on the influence of RPL parameters and network characteristics on the network formation time is presented in [180], using OMNeT++ [181] as simulator. In our work, we use a similar methodology (see Section 4.3.2), however considering different network scenarios and only a fixed value for the RPL redundancy threshold. The limitations of a strict separation between the IEEE 802.15.4 MAC and routing protocol RPL, especially during the formation of a multi-hop wireless network, are outlined in [182]–[184]. These three references prove the advantages of using a cross-layer optimisation. Further, Pavkovic et al. [182] and Vucininic et al. [183] address the incompatibility of the IEEE 802.15.4 cluster-tree and the DODAG. To this end, a new cluster-DAG structure for IEEE 802.15.4 is described in [182] and an interplay scheme between IEEE 802.15.4 and RPL during network formation is proposed in [183]. Iova et al. [184] recommend among other things that the MAC protocol does not impose a topology to the network, but only filters out bad links. The 6TiSCH standardisation activities have followed this recommendation in defining the 6TiSCH-MC [101] (see Section 4.2), where the allocated single shared TSCH cell is a sort of

broadcast channel, adopted by all node for constructing the RPL topology without any restrictions imposed at MAC layer.

3.2.3 6TiSCH Network Formation

While a considerable amount of literature has been published on designing scheduling algorithm for delivering data packets in an operative 6TiSCH network, the initial stage of such networks has been less investigated. Remarkably, few works consider the interplay of the different protocols forming the 6TiSCH protocol stack during the network formation process.

Duquennoy et al. [185] define Orchestra, which can be considered as a breakthrough autonomous scheduler. Although Orchestra is mainly a scheduling solution that aims for high reliability in transmitting unicast data messages, it might also be useful in the network bootstrap phase. Indeed, the Orchestra scheduler might set slotframes to exchange solely control packets or for basic connectivity between nodes. For example, the Minimal Scheduling Function (MSF) [186], which is currently under standardisation, adopts Orchestra rules for the distributed resource management at network bootstrap. Nevertheless, the authors of Orchestra do not focus on this initial stage of a 6TiSCH network; instead, they study the performance of the scheduler in a stable formed 6TiSCH network. Wang et al. [187] examine how quickly a joining node synchronises to an existing TSCH network when the 6TiSCH-MC is used. They present results achieved by simulations with Cooja, and they highlight the impact of the EB period, and neighbourhood size on the synchronisation time. Our performance study is a sort of extension of this reference work, considering the more extensive set of configuration parameters and topologies that we have simulated in our experiments. Righetti et al. [188] evaluate the performance of the 6top Protocol (6P), defined in 6TiSCH, considering the exchange of the messages to allocate dedicated timeslots at bootstrap. As in our contribution, this work (i) gives guidelines on tuning TSCH parameters, e.g. at least two shared timeslots for a successful allocation by each node, and (ii) considers the interplay between TSCH and RPL. The limit of the 6TiSCH-MC in dense networks is considered as a question of fact by Vucinic et al. [189] and motivates in their work the proposal of a 6TiSCH specific network advertising strategy as an answer to this problem. The authors present a Bayesian broadcast algorithm which optimises the 6TiSCH network formation process by setting (locally and dynamically) different transmission probabilities for each type of control traffic, i.e. EB, DODAG Information Object (DIO) or 6P negotiation messages. In contrast to our solution, this approach is compliant with 6TiSCH-MC. However, the authors do not investigate the issue related to the

interaction of TSCH and RPL protocol parameters and the configuration of these parameters so extensive as we do in Section 4.3.

Very close to our contributions are the references [190], [191]. Through Cooja simulations, Vera-Pèrez et al. [190] investigate the optimal configuration of the TSCH and RPL parameters for improving the 6TiSCH network formation. In particular, the authors suggest that nodes adopt a short period (i.e. $4\,\mathrm{s}$) during the first two minutes of network advertising and then a longer period (i.e. $16\,\mathrm{s}$) for transmitting EBs. Vallati et al. [191] demonstrate through both an analytical model and simulations that the 6TiSCH-MC with only a shared cell for control traffic might lead to poor performance in term of network formation time and routing functionalities. Therefore, the authors propose a dynamic scheduling algorithm to control the allocation of additional shared cells. Both works confirm and complement our performance evaluation on the shortcomings of the 6TiSCH-MC. Similarly to [191], we suggest to allocate the cells for broadcasting control message equally spaced within the slotframe; still, we additionally investigate with our extension of the 6TiSCH-MC if a coordinated reservation of these cells among nodes might lead to significant performance improvement.

3.3 Related Work on LQE in 6TiSCH Networks

Recall from Section 2.3.1 that low-power wireless links are per nature unreliable and their quality may significantly fluctuate over time. Enabling a reliable and energy-efficient communication in WSNs is a matter of selecting the wireless links with better quality at the time of transmission. However, nodes can not have an always up-to-date knowledge about the link quality to the available neighbours. The viable strategy is to observe the wireless links' behaviour and estimate their quality over a certain time horizon. Hence, LQE has been established as term for the process that gives a statistical characterisation of the links. We explain this process, its specific requirements and most representative LQE metrics in Section 3.3.1.

LQE plays a crucial role in many mechanisms or protocols adopted in WSNs. For instance, both at MAC and routing layer, the topology construction relies on the statistical characterisation of links to (i) restrict the set of nodes that are considered neighbours of a given one or (ii) establish the best available multi-hop path (recall RPL DODAG construction in 2.4.4). In Section 3.3.2, we look more closely the different LQE mechanisms that exist for RPL. At the same time, LQE can support the building of communication schedules in TDMA-based WSNs. For example, the

scheduling function can use the link quality for reserving the right amount of transmission slots, and, thus, providing latency and reliability guarantees [192]–[194]. In addition, when channel hopping is adopted as in TSCH or similar protocols, a LQE for the different frequencies allows using only the subset of the best frequencies for data transmissions. In this context, the term Channel Quality Estimation (CQE) is the keyword in many research papers. We review the state-of-the-art of CQE methods in 6TiSCH networks in Section 3.3.4, where we also position our proposal presented in Chapter 5 within the related literature.

3.3.1 Link Quality Estimation in WSNs

The LQE is a mechanism that a node implements to gather and maintain up-to-date information about the quality of the links to its neighbours. Specifically, LQE is a process that consists of the following three components [43], [44]:

- *Link monitoring*, which is the strategy used to collect link measurement during an observation window $[t_0, t_1]$. We can discern between *active* and *passive* strategies. In the former case, a node transmits specific (either broadcast or unicast) probe packets to the neighbours to monitor the link actively. In the latter case, a node retrieves measurements passively, exploiting only the existing traffic (both data packets and their acknowledgement) to infer the link quality. Moreover, *hybrid* approaches, which combine both strategies, are possible.
- *Link measurement*, which specifies the data to be retrieved. Basic link measurements can be taken directly from the radio transceiver (*hardware indicators*). Besides, the MAC layer can provide counter-based indicators by storing the number of packets successfully received (or delivered and acknowledged), thus, characterising the link quality towards a specific neighbour node. In addition, we can distinguish between *receiver-side* and *sender-side* link measurements according to the art and source of the retrieved data.
- *Metric evaluation*, which defines the metric and the estimation method to assess the link quality over a particular time horizon $[t_1, t_2]$. The metric may be a single hardware or counter-based indicator or either a more complex combination of such indicators that combines their strengths. The estimation technique defines the observation windows' size and how the collected samples are filtered to smooth out the short-term fluctuations.

Tab. 3.1: Classification of low-power wireless links, from [42], [43].

Category	LQI	SNR	PRR
Very good link	106+	30+	1
Good link	102 - 106	15 - 30	0.75 - 1
Intermediate link	80 - 102	5 - 15	0.35 - 0.75
Bad link	0 - 80	0 - 5	0 - 0.35

Several efforts were carried out for designing a link quality estimator that is *accurate* (i.e. reflecting the real link behaviour), *adaptive* (i.e. able to quickly detect significant variations in link conditions), *highly efficient* (i.e. requiring low processing, communication and memory overhead) and *stable* (i.e. able to tolerate short-term fluctuations) [43], [44]. These desirable properties are conflicting, and it is still challenging to find the right trade-off between them. For example, a statistically meaningful metric (i.e. accurate LQE) demands communication and processing overhead that can cost precious energy or be awkward with the limited computational capability available in constrained devices.

In the following, we briefly describe the most representative LQE metrics for WSNs.

Simple LQEs use directly (or the average of) one of the following the hardware or counter-based indicators. The **Received Signal Strength Indicator (RSSI)** and **Link Quality Indicator (LQI)** are delivered as register values from any IEEE 802.15.4-compliant transceiver. From them, one can easily derive the **Signal-to-Noise-Ratio (SNR)**. Other two simple observable quantities are **Packet Reception Rate (PRR)** and **Packet Delivery Rate (PRR)**. Table 3.1 shows a possible classification between *very good*, *good*, *intermediate* and *bad* links using this set of basic metrics, as suggested in references [42], [43].

To improve the performance of the simple LQEs presented above, more complex processing algorithms have been investigated [195], [196]. An **Exponentially Weighted Moving Average (EWMA)** estimator has been found to give the best compromise between stability and agility. It computes a prediction P_n at time $t_n = t + nT$ about an indicator μ (i.e. RSSI or PRR) and its measured value at instant n, i.e. μ_n, as:

$$P_n = \alpha \cdot P_{n-1} + (1 - \alpha) \cdot \mu_n \tag{3.1}$$

where the factor $\alpha \in [0, 1]$ controls the effect of the previously computed values to the new one.

A widely used and well-acknowledged metric is **Expected Transmission Count (ETX)** [119]. It computes the expected number of packet transmissions required to

successfully deliver a packet on a link. The ETX of a link is calculated using both the forward (d_f) and reverse (d_r) delivery ratios of the link, thus handling asymmetric loss ratios, as:

$$ETX = \frac{1}{d_f \cdot d_r} \tag{3.2}$$

As a routing metric, ETX is designed to optimise throughput, latency and energy consumption in the network [119]. However, it is not the best when it comes to end-to-end reliability [197]. Moreover, there are different approaches to measure both delivery ratios d_f and d_r . The authors of [119] suggest using small dedicated broadcast probe packets (i.e. an active strategy), but other works adopt a passive measurement technique to save energy. For instance, the authors of [198] use only the existing unicast data traffic and defines the ETX as the ratio between the number of acks received and the number of data packets sent on that link, including retransmissions.

Differently from ETX, which assesses only the number of packet retransmissions over the link, **Fuzzy-LQE** [199] (or its optimised version Opt-LQE [200]) is a composite metric that estimates the link quality considering four properties, namely PRR, SNR, ASL (link ASymmetric Level) and SF (link Stability Factor). Fuzzy-LQE combines these properties together using Fuzzy Logic, providing a score ranging from 0 to 100 (where 0 is the worst quality, and 100 is the best). A comparison of Opt-LQE and ETX as routing metrics in terms of reliability, communication delay and energy efficiency is made in [201], and it suggests a superiority of Opt-LQE. However, in our opinion, this conclusion mainly originates from the particular ETX implementation (i.e. default passive strategy), which the authors adopt for the comparison in their work. As we also explain later (see Section 5.3), active probing is a key strategy for improving the accuracy and stability of the ETX metric, and, thus, assisting routing decisions [202].

Four-Bit (4B) [203] is a cross-layer metrics which encodes the link quality in four bits of information obtained from multiple layers: one from the Physical (PHY) layer (about the quality of received signals), one from the MAC layer (about the number of acknowledged packets) and two from the NET layer (about the relevance of link for routing). However, this cross-layer approach produces a link behaviour that is complicated to understand, and the produced metric tends to be unstable as the variations at each layer are amplified when combined together [198].

Quality metrics that give a worst-case link description are useful for building TDMA schedules for reliable and timely data delivery in WSNs. A representative example of such metrics is **Bmax** [192], [204]. The metric Bmax expresses the link burstiness, i.e. the maximum number of consecutive failed transmission on a link. It is computed

using a training phase, in which a large number of probe packets are transmitted on the link for collecting statistics about packet losses. However, the LQE process for evaluating this metric on each available link in a network is quite long (more than $140\,\mathrm{h}$, according to the authors). Further, the training phase has to be restarted from scratch each time the radio environment changes.

In Chapter 5, we analyse the performance of existing passive and active monitoring strategies for LQE. We highlight that those strategies, which provide accurate link metrics with traditional CSMA/CA-based MAC protocols, do not perform well in 6TiSCH networks, especially during their crucial initial phase. The estimation of the well-established metric ETX results to be inaccurate due to the specific notion of link in a 6TiSCH network (see Section 5.2). This problem would also affect the different metrics based on ETX proposed in the past years, which we briefly describe in Section 3.3.2.

3.3.2 LQE in RPL

Given that the RPL specifications [73], [205] do not explicitly define an LQE mechanism, a large volume of published studies addressing an accurate and stable LQE in RPL networks exists.

Some of these studies investigate the efficiency of RPL performance under different link quality metrics. The impact of widely used metrics on the reliability, the end-to-end delay and energy efficiency of RPL has been evaluated by the authors of [198](through simulations with Cooja) and [206](through a testbed). Both studies do not recognise any metric among RSSI, LQI, 4B, Fuzzy-LQE and ETX that outperforms the others in every performance aspect or scenario. Thus, our choice of selecting ETX in this thesis is not a wrong decision; actually, it is the recommendation for dense networks [206]. Although Iova et al. [198] observe an instability problem in the RPL structure when using ETX, we argue that this finding comes from the initialisation phase and their setting $ETX_{Init} = 1.0$, an extremely non-conservative preferred parent solution, as also pointed out in [207]. In contrast, in our study, we choose $ETX_{Init} = 5.0$ as the unknown quality of any link. This is a compromise between the lowest and the greatest possible values in the network.

Other researchers have attempted to define novel routing metrics for RPL in specific scenarios or applications. We restrict our attention to those works that have suggested either (i) alternative methods for computing ETX or (ii) a composite metrics based on it. Iova et al. [184] recommend to compute ETX by taking into account also

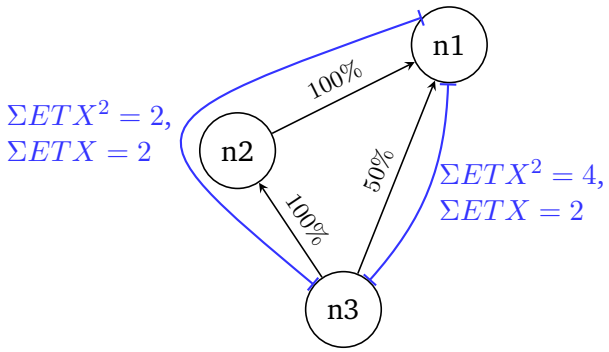


Fig. 3.1: Using $ETX = 1/PDR$ as routing metric, RPL can not distinguish between the single-hop link $n3 \rightarrow n1$ and the path $n3 \rightarrow n2 \rightarrow n1$, as both paths have the same accumulated ETX value. Instead, using $ETX^2 = (1/PDR)^2$ as routing metric, RPL selects the more reliable path, which consists of two perfect links.

the packet dropped at the MAC layer due to successive Clear Channel Assessment (CCA) failures. Note that this recommendation loses its importance in 6TiSCH networks; indeed, TSCH does not adopt CCA to prevent collisions among nodes (even in shared cells) and are optional [109]. Duquennoy et al. [197] introduce a variant of ETX, namely ETX^N, to favour reliable link for building multi-hop paths in RPL. ETX^N raises the traditional ETX (i.e. merely the inverse of the PDR) to the power of a number N. Figure 3.1 depicts an example with $n = 2$ and reveals how RPL using ETX^2 considers a path with two reliable links better than a single-hop link with a PDR of $50\,\%$, whereas both paths have the same (accumulated) cost with the traditional ETX. Nevertheless, the benefit of using ETX^N would be petty without an accurate estimation of ETX in 6TiSCH-network, which is our main contribution in Chapter 5.

So far, this section has described the relevance of ETX (or its variations) as a primary routing metric for RPL. In addition, ETX has been combined with other metrics in RPL to meet different application requirements. For instance, in energy-aware routing, two examples of composite metrics based on ETX are Expected Lifetime (ELT) [74] and Lifetime and Latency Aggregable Metric (L^2AM). Both proposals aim at prolonging the overall network lifetime from both an ETX and an energy operation point of view. For improving the robustness of WSNs against malicious or unreliable nodes, the authors of [208] combine ETX with a trust metric. Another example of ETX-based composite metrics is the Neighbourhood Metric (NM) [209], which results by summing a specific score for each neighbour with the ETX of the link. Thus, NM provides the opportunity of including information on the quality of a node's neighbourhood. All these composite metrics would surely benefit from the accurate ETX offered by our proposal for 6TiSCH network.

As already introduced in Section 3.3, a prerequisite for LQE is the link monitoring to collect the measurements needed for the metric evaluation. We can find several examples of *active* and *passive monitoring* strategies adopted for the routing protocol RPL and the ETX metric's computation in the literature. On the one hand, a node can passively infer the link quality through existing unicast data traffic with the preferred parents [207] or RPL control packets DIO [198], [210]. Also overhearing transmission from neighbouring nodes [211] falls into the passive approaches. On the other hand, the active probing strategy injects particular probe messages to collect link quality information. Nodes do active probing either with a specific periodicity [197] or with a burst of packets triggered by certain events (e.g. during parent changes) [210]. Moreover, probe packets can be either sent as unicast to a specific neighbour [197], [212] or broadcast [213]. Probe broadcasting results in a smaller overhead than unicast probing since one probe helps to refine the link quality towards all the neighbours. Vallati et al. [202] have carried out an experimental study comparing the active and passive probing strategies available in the RPL

Besides the basic measurements schemes described above, there are some examples of more sophisticated link-quality measurement frameworks for RPL. For instance, Ancillotti et al. [212] have proposed a hybrid probing scheme that combines both passive and active technique and selects one of them depending on RPL operations and on characteristics of the link to be monitored. In [214], the authors have proposed a novel strategy called RL-probe for RPL, which combines synchronous (i.e. active probing with unicast packets) and asynchronous (i.e. similar to the above cited [210]) LQE techniques. The RL-probe splits neighbouring nodes into separate groups, and it adopts a reinforcement learning scheme for determining which group to probe and how frequently. However, the authors have designed RL-probe explicitly for single-channel MAC protocols; thus, its application in multi-channel 6TiSCH network is still an open research issue. Gomes et al. [215] have proposed the deployment of specific LQE nodes in the WSN to monitor the link quality in industrial environments in real-time, using the RSSI and information obtained from the received data packets. Although this approach does not explicitly require the generation of probes, it demands a more significant number of deployed nodes, thus increasing the deployment cost.

It is important to point out, that all the studies mentioned above assume IEEE 802.15.4-2011 (or other precedent versions) either with CSMA/CA [74], [184], [198], [216] or with ContikiMAC as radio duty cycling mechanism [201], [212], [214]. The LQE mechanisms proposed for RPL in those works do not obviously perform in the same way, when at the MAC layer the TSCH protocol is adopted. In particular, the time-slotted access mode and the channel hopping technique complicate the

notion of link as intended for CSMA/CA like protocols. We explain this problem in Section 5, where we show the fallacy of adopting popular LQE strategies on 6TiSCH networks.

3.3.3 LQE for 6TiSCH Networks

When considering 6TiSCH networks (i.e. TSCH at MAC layer together with RPL as routing protocol), to the best of our knowledge, there are only two works on LQE. In [217], the authors have proposed a passive LQE mechanism for ranking neighbouring nodes using the rate of broadcast packets received, such as EBs and DIO messages. The basic premise behind this proposal is the demonstrated correlation between the reception rate of broadcast packets from a given neighbour and the unicast link quality to it. Although this solution filters out bad neighbours and reduces the number of parent changes [64], its performance strictly depends on the rate at which broadcast packets are transmitted and on their distribution among all the channels. In the study, the authors have disabled the RPL Trickle algorithm for getting a fixed broadcast period and for avoiding the decrease of the DIO emission rate, which is imposed by Trickle as the network stabilises. Besides, the transmission of EB and DIO messages happens in shared cells, which are prone to collisions. The occurrence of collisions in dense networks may lead to inaccuracy by ranking the neighbours, and it is the primary motivation for the adoption of dedicated cells for probing in our proposal. Recently, Henrique-Gomes et al. [218] have suggested with TAMU-RPL an optimisation of RPL for improving its reactiveness to link degradation. A key algorithm in TAMU-RPL is an LQE mechanism based on Thompson-sampling heuristic, which enhances the ETX estimation to a subset of neighbours by exploiting the current best parent. Additionally, TAMU-RPL supports multi-channel MAC protocols (as TSCH) measuring the ETX per channel and using RSSI information from all unicast and broadcast packets that are overheard. Although this solution was designed for multi-channel LLN, in our opinion, its suitability to 6TiSCH networks is still questionable. Indeed, the authors have evaluated TAMU-RPL using an atypical TSCH slotframe, which is composed of only shared slots and leads to a nearly multi-channel version of CSMA/CA. Besides, its energy-expensive overhearing mechanism may even be impossible in a typical 6TiSCH network when different frequency are used in the neighbourhood during a specific timeslot.

3.3.4 Channel Quality Estimation for TSCH

Most TSCH implementations adopt a blind channel hopping mechanism, i.e. they use all available 16 frequencies. Nevertheless, the reliability in a TSCH network can be further improved by selectively avoiding low-quality channels, as pointed out in [69]. This technique is known as blacklisting or whitelisting [1]. It relies on channel quality estimation (CQE), i.e. the process of estimating the quality of each of the 16 channels, which can be used on the link between two neighbouring nodes. In contrast, LQE ignores the single channel quality conditions on a specific link. However, the challenge is to find a practical CQE implementation which has a cost smaller than the benefit. First, CQE should select k channels out of 16, a more resource-demanding task than merely selecting the best one (as in Zigbee) [219]. Besides, the appropriate value for k depends on the environment (i.e. how congested the $2.4\,\mathrm{GHz}$ band is), and it should not excessively reduce the network capacity. Second, accurate CQE requires active and frequent measurements with probes or additional dedicated nodes for this task. Both approaches involve not negligible additional energy or financial resources [220]. Finally, neighbouring nodes must agree on the blacklisted (or whitelisted) channels, and, thus, they have to periodically exchange hopping sequence information. This complicates the applicability of blacklisting in multi-hop or dense TSCH networks.

An adaptive channel hopping mechanism is investigated in [221]. Each node primarily uses a selected channel c_i for data communication, but it monitors and updates the Packet Delivery Ratio (PDR) of each other radio channels c_j by switching to c_j periodically and shortly (e.g. for one packet transmission every *k* ones). However, the authors themselves admit only a marginal benefit against the blind channel hopping. The authors of [222] reserve two network-wide idle TSCH-cells (called *noise floor* slots) in each slotframe, where each node samples the noise levels of the channels through RSSI measurements (recall previous Section 3.3). The Non-Intrusive Channel quality Estimation (NICE) technique proposed by Tavakoli et al. [223] enhances the latter method by performing energy detection more frequently, i.e. exploiting the idle periods offered by TSCH in every timeslot. Elsts et al. [219] investigate the effectiveness and the overhead of RSSI- and PRR-based metrics for assessing the channel quality and, thus, driving adaptive channel selection. The authors of [224] and [225] formulate this process as a multi-armed bandit (MAB) problem [226]. Thus, both contributions realise a CQE strategy that provides the best trade-off between exploring the unknown environment (i.e. all channels) and

[1] Both blacklisting and whitelisting are used interchangeably in the literature. Blacklisting constructs the list of channels that must be avoided for transmissions; oppositely, whitelisting refers to the subset of channels that form the channel hopping sequence

exploiting the current knowledge (i.e. the selected subset of good channels). Each node executes the MAB algorithm for all neighbouring nodes. However, this process requires large intervals to converge network-wide, a fact that explains its evaluation with a fixed routing tree and a static schedule in [225] or with a star topology in [224].

In addressing CQE in TSCH-network, most of the works mentioned above, namely [219], [222], [223], use EWMA filtering (c.f. Section 3.3) to generate a stable estimate of the channel qualities. We also adopt this exponential smoothing technique for an accurate and stable ETX in our proposal. Further, all these related works apply to the operational stage of a TSCH network, when both the topology and the communication schedule are built and fixed. They aim at reducing the number of retransmissions by adapting the channel hopping sequence on already established links. Instead, our proposal plays a crucial rule already before this operational state of the network, because it helps the routing and scheduling protocol in filtering out bad neighbours (i.e. links with low quality) and in constructing efficient routes for improving the network reliability. Further, we assume scenarios with multi-hop networks; instead, the focus in the related works on CQE for TSCH is on star or fixed tree topologies. For our work, we choose not to assess the channel quality along with the LQE strategy. Therefore, when a node receives probe messages from a neighbour, it does not differentiate the reception rate per channel. It follows that our LQE method averages the ETX over all the available channels. The rationales behind our choice of refusing channel quality assessment are that (i) the memory overhead for maintaining LQE statistics in each node remains small, (ii) the distribution of LQE statistics inside a probe packet requires few extra bytes and (iii) the applicability of our LQE strategy during the network bootstrap phase is not compromised.

3.4 Chapter Summary

In this chapter, we provided a literature review to contextualise the research conducted in this thesis.

First, we focused on the works that address the initial phase of a multi-hop WSNs and the mechanisms for network advertising, joining and topology construction. At the MAC layer, it is necessary to coordinate the beacon transmission among advertiser nodes and avoid their collisions. Classical 802.15.4 CSMA/CA networks have already addressed this problem one decade ago but on a single radio frequency and introducing a hierarchical structure for the multiple coordinators. Therefore, these

solutions are not directly applicable in 6TiSCH networks. This chapter reviewed different beacon scheduling algorithms for TSCH, which were proposed in the literature so far. We distinguished between solutions that dynamically adapt the beacon generation rate and strategies for allocating advertising cells in the TSCH slotframe. In our work (see Chapter 4), we follow the latter approach and we propose an algorithm that dynamically reserve these cells considering the node density and the experienced collisions. Further, this chapter pointed out that the interactions between the MAC and NET layers, i.e. TSCH and RPL in 6TiSCH, have attracted little attention so far. However, the relationship between TSCH and RPL protocol parameters has a significant impact on the performance of the bootstrap phase, as confirmed by our investigations presented in Chapter 4.

Next, we explained why estimating the link quality is of prime interest for WSNs. Low power wireless links are unreliable and dynamic per nature. Routing decisions and communication schedules aiming at latency or reliability guarantees rely on the statistical characterisation of links that LQE offers. We described the most representative LQE metrics for WSNs. We gave special attention to the ETX metric and the mechanisms for its computation that have been proposed for RPL in literature. We mentioned that those strategies do not perform well in 6TiSCH networks, especially during the network formation phase. This fact mainly motivated our investigations. Further, we highlighted how many previous works can benefit from the accurate ETX offered by our proposal presented in Chapter 5

In the next chapter, we start presenting the main contributions of our work.

4 Improving 6TiSCH Network Formation

4.1 Chapter Overview and Contributions

This chapter focuses on the initial deployment stage of 6TiSCH networks, which is characterised by synchronising nodes on a common Time Slotted Channel Hopping (TSCH) slotframe (recall Section 2.4.3) and building an Routing Protocol for Low Power and Lossy Networks (RPL) Destination Oriented Directed Acyclic Graph (DODAG) structure (recall Section 2.4.4). These two operations must occur before any node can start transmitting data packets and mainly rely on properly flooding control messages. For this purpose, the 6TiSCH WG has proposed the *6TiSCH Minimal Configuration (6TiSCH-MC)* [101]. In particular, this RFC document specifies a static resource allocation for exchanging every packets during the network formation phase. In this chapter, we first highlight how the blind adoption of 6TiSCH-MC may produce a very long (or even unsuccessful) network formation phase, mostly due to the suggested static resource allocation and the recommended configuration of TSCH and RPL. It is important to highlight that a prolonged network formation time matters in industrial applications. On one side, as long as a node has not joined the network, it can not provide information regarding the monitored process. On the other side, it may impair the operational lifetime of the network; indeed, the initial TSCH synchronisation process may consume an unreasonable amount of energy for battery-powered devices, when the joining nodes keep the radio always on for a considerably long time, waiting for the first synchronisation message. Therefore, we evaluate through simulations the delay and the energy cost with the 6TiSCH-MC before getting an operational network. The goal is to derive a set of guidelines for an appropriate parameter setting of the protocols in the 6TiSCH stack, depending on some typical network topologies.

From this evaluation, we find out that reserving additional transmission resources (i.e. active shared cells in the TSCH slotframe) is a viable approach that is worth to be explored. Indeed, this strategy can ensure a more reliable delivery of control messages, which means a shorter 6TiSCH joining time by each node and less energy consumed during the bootstrap operations. Nevertheless, such a simple strategy

may produce a waste of bandwidth at steady state [227] or require management overhead for specifying the resource allocation. Besides, a question that needs to be answered is how to allocate these active cells, both in the TSCH slotframe and among nodes.

We investigate in this direction by defining a dynamic allocation strategy, called *Multiple Advertisement Cell Allocation (MACA)*, for coordinating the scheduling of several advertisement cells, which each node uses to transmit and receive the control messages that enable the initial TSCH synchronisation and the DODAG formation. The main goal of MACA is to significantly reduce the collisions of bootstrapping traffic. We achieve this goal by encapsulating in the Enhanced Beacon (EB) information that nodes use for learning how many active cells are instantiated in the neighbourhood and which of them can be selected for advertising without causing collisions. We evaluate the performance of the proposed MACA procedure in-depth and compare it against other two straightforward extensions of the 6TiSCH-MC, which are closely comparable to the related works [109], [170], [227], [228].

In short, the main contributions of this chapter are:

- The investigation of the 6TiSCH network formation process and the interplay between TSCH and RPL during this process;
- An extensive simulative study of the standardised 6TiSCH-MC that highlights its limits and provides a set of guidelines for an appropriate setting of TSCH and RPL parameters;
- The presentation of MACA, a new dynamic allocation strategy for improving the 6TiSCH network formation;
- A performance evaluation of MACA and other straightforward extensions of 6TiSCH-MC in different challenging scenarios.

The remainder of this chapter is structured as follows. We give a detailed description of the 6TiSCH network formation process in Section 4.2. Section 4.3 describes our extensive simulative study of the 6TiSCH-MC. There, we discuss the shortcomings of this minimal mode of operation standardised by the 6TiSCH-WG, and we give guidelines for more appropriate parameter settings. Then, Section 4.4 describe the principal elements of the MACA algorithm, i.e. our decentralised solution for coordinating the allocation of several advertisement cells in the minimal slotframe among the joining nodes during the network bootstrap. Further, we present a performance evaluation of the proposed algorithm, and also a comparison between

the MACA algorithm and other two straightforward extensions of the 6TiSCH-MC. Finally, Section 4.5 summarises the chapter.

4.2 Standardized 6TiSCH Network Formation Procedure

Any 6TiSCH compliant device has at least to synchronise on a slotframe structure and to join the logical routing topology before it can send data messages to the sink node. We call these two actions *TSCH synchronisation* and *RPL DODAG joining*, respectively. In particular, the TSCH synchronisation assures a reliable single-hop communication between neighbour nodes, whereas the RPL DODAG joining the multi-hop data forwarding toward the sink.

The ongoing execution of these two actions in each node defines the 6TiSCH network formation process. It starts from the sink node, which specifies the network setting and is the first device that advertises the network presence for other nodes. It concludes when every node has joined both the TSCH network and the RPL DODAG topology, i.e. the *TSCH network-wide synchronisation* and *RPL DODAG construction* are attained.

In this initial deployment phase, advertising the network presence is a fundamental primitive, both at Medium Access Control (MAC) and at Network (NET) layer. To this end, IEEE 802.15.4-2015 specifies the Information Element (IE) frame and its format for the TSCH mode. As detailed in Section 2.4.3, an EB contains all the necessary time information to allow the initial TSCH synchronisation among nodes, including the timeslot duration, the slotframe length and the channel hopping sequence. However, the standard does not detail a specific transmission strategy for EBs; instead, it only suggests that nodes, after their synchronisation, generate an EB within an interval t_{eb} and broadcast it with the TSCH channel hopping mechanism, i.e. on different frequencies at different slotframes. Similarly, the RPL protocol defines DIO packets (recall Section 2.4.4) for advertising the DODAG-ID and other configuration parameters required for the DODAG construction. However, the generation of DIO messages does not happen at a fixed rate as for EBs; instead, the RPL protocol adopts the Trickle algorithm [229], which variates the frequency of DIO transmissions using the following logic. As long as RPL observes a stable and consistent DODAG topology, the Trickle algorithm doubles the DODAG Information Object (DIO) interval size I, starting from the minimum size I_{min}, until it reaches a maximum value $I_{max} = 2^M \cdot I_{min}$. If, however, an inconsistency in the DODAG

is detected, the Trickle algorithm resets I to the minimum size I_{min}. In both case, the transmission of DIO messages is scheduled at a random instant τ in the second half of the interval I. Thus, the Trickle algorithm allows, on one side, to reduce the quantity of DIO messages in a stable routing topology; on the other side, to broadcast these control messages as quickly as possible in case of changes in the topology.

As highlighted above, the 6TiSCH network formation is a dynamic process that involves two different layers (MAC and NET). Further, the configuration of parameters or the definition of some mechanisms and rules in TSCH and RPL protocols (e.g. EB advertising strategy, Trickle reset events) are left open to implementers. As expressed in [184], in this context, there is the risk of shortcomings due to blind layer separation, but also an ample opportunity for cross-layer optimisation. For these reasons, the Internet Engineering Task Force (IETF) 6TiSCH Working Group (WG) has recently been working on the definition of a set of conventions to build the network, assuring essential interoperability between nodes, and defining how TSCH should interact with RPL during the network formation. These standardisation efforts have been materialised into the so-called *6TiSCH Minimal Configuration (6TiSCH-MC)* [101], which specifies in particular (i) a static TSCH-schedule for the whole network for the transmission of bootstrapping traffic and (ii) the guidelines for the coordination between the *TSCH network-wide synchronisation* and *RPL DODAG construction*, as explained below.

The static schedule defined in 6TiSCH-MC is a *minimal slotframe*, consisting of a single active cell, used to transmit/receive all the control traffic in a slotted Aloha manner. This cell is located at timeslot offset 0 and channel offset 0. The slotframe length L_{mc-SF} is tunable to trade off bandwidth for energy consumption. The recommended value $L_{mc-SF} = 101$ results in a radio duty cycle below $0.99\,\%$.

As already mentioned previously, the sink node starts the 6TiSCH network formation process. In doing that, the sink node carries out two distinct roles: the TSCH coordinator and the RPL root. As TSCH coordinator, it specifies the MAC layer configuration and broadcasts EBs. As RPL root, it emits DIO packets.For broadcasting EB and DIO packets, the sink node exploits the shared slots scheduled every $L_{mc-SF} \cdot t_s$ through the *minimal slotframe*. This process is exemplarily shown in Figure 4.1, where $t_{eb} = 200\,\mathrm{ms}$, $I_{min} = 128\,\mathrm{ms}$ and $L_{mc-SF} = 11$ are used. After being generated, EB and DIO messages are put in the transmit queue and processed by TSCH in the next active slot. As illustrated, EBs are queued with a priority higher than DIO messages.

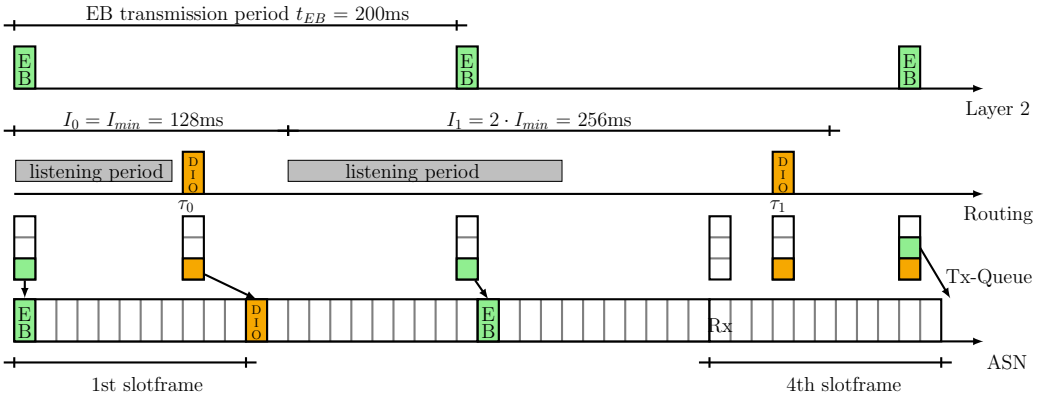


Fig. 4.1: A timeline of the network formation on an advertiser node

When a node wishes to join the network, it preferably uses passive scanning, i.e. it turns the radio on to a randomly chosen frequency, and it listens for EBs. While waiting for a valid EB, the joining node keeps its radio always on and changes frequency every t_{scan}. After hearing a valid EB, a node learns the *minimal slotframe* in the network (i.e. Absolute Slot Number (ASN), the timeslot duration, slotframe length, number of available frequencies), so it exactly knows when to wake up for receiving or sending control messages related to the network formation procedure. The joining nodes will not start advertising the network presence straight away, but only after it has received a DIO message, computed its rank and selected its preferred parent among a set. Further, the preferred parent of the joining node becomes its time source neighbour. With this expedient, the 6TiSCH-MC matches the link-layer topology and the routing topology. Hence, it assures a network-wide time synchronisation with a loop-less structure (offered by the RPL DODAG topology) and with no extra control messages at MAC layer [61]. Optionally, a newly synchronised node can multicast a DODAG Information Solicitation (DIS) to ask for a DIO and to speed up the joining.

This process allows to gradually cover the whole network: each node may become an advertiser node for other nodes that are not in the transmission range of the sink one. All control traffic listed above (i.e. EB, DIO and DIS messages) is transmitted within the single shared cell provided by the *minimal slotframe*. Therefore, as more and more nodes join, collisions are likely to occur.

Our description of those two interacting mechanisms illustrates that answering the questions *how long the network formation process will take?* and *how the TSCH and RPL protocol parameters should be set?* is not trivial. First, the multiple available physical frequencies and the channel-hopping mechanisms of TSCH improve the robustness of network advertising against external interferences but, at the same

time, they lead to a longer synchronisation time at each joining node. Indeed, the joining node and the advertiser node need to be aligned on the same frequency when the latter sends the EB frames. Next, the number of deployed nodes and their density has undoubtedly an impact on the network formation time. Due to the progressively joining of nodes in the 6TiSCH network, we can expect that each additional hop in the routing topology costs extra time to have an operational network. Nevertheless, expanding the network size implies increasing the number of advertiser nodes involved in transmitting control messages. On one side, this fact might reduce the synchronisation time of nodes that desire to join the network, since the probability of receiving a bootstrap message increases [169] when more advertisers are in the neighbourhood. On the other side, collisions among control message occur more likely, so that a policy for scheduling EBs and DIO messages might be mandatory. Finally, there is an interplay between TSCH and RPL protocols and their (default) parameters in every phase (scanning for EBs, synchronising, joining a DODAG). Since 6TiSCH-MC gives DIO messages a lower priority than EBs, the former may be significantly delayed or even dropped with an improper setting of the Trickle parameters I_{min} and M that ignores the TSCH slotframe length or the EB interval period t_{eb}.

For these reasons, the next Section 4.3 presents an in-depth simulative study on the impact of different 6TiSCH parameters and other topology factors on the duration and the energy consumed for network formation process when the 6TiSCH-MC's recommendations are adopted. In particular, we draw attention to the interaction between TSCH and RPL in this initial phase.

4.3 Network Formation with 6TiSCH-MC:Limits and Recommendations

As explained above, the 6TiSCH-MC defines the basic set of rules and parameters for operating a 6TiSCH network. More, it enables conformance testing of a device to the standard and interoperability between multiple devices from different vendors [230]. However, the 6TiSCH-MC is not a one-size-fits-all solution that implementers of Industrial Internet of Thing (IIoT) solutions can blindly use. For instance, when they deal with relatively large topologies, the common and basic configuration of the protocols in the 6TiSCH stack leads to unsatisfactory performance at bootstrap. This shortcoming also impairs the network performance in the operational phase (e.g. not operational nodes, reduced energy budget). To quantify this, we present

hereafter our extensive simulative study of the network formation process proposed with 6TiSCH-MC [101]. We evaluate the delay and the energy cost in this phase. Implementers of IIoT solutions will benefit from our guidelines for an appropriate parameter setting, depending on some typical physical topologies.

4.3.1 Methodology

We present here the methodology used to evaluate the network formation process and our principal findings. For reasons already explained in Section 2.2.3, we evaluate the performance of the initial 6TiSCH network formation using Cooja.

In the simulations, we use the Cooja mote type, i.e. a virtual hardware without limitations concerning memory and computation capabilities. As wireless propagation model, we use the Unit Disk Graph Medium (UDG). Links are symmetric, the packet delivery ratio (PDR) is $100\,\%$ in a transmission range of $50\,\mathrm{m}$, and the interference range is $100\,\mathrm{m}$. We did not consider a more realistic hardware and radio setting to narrow down the side effects and to study the protocol mechanisms alone.

Similarly to [188] and [180], we consider different network sizes ($N_{size} \in \{9, 16, 25\}$) and three categories of network topology:

- *Grid*: the sink is placed in the upper left corner of the grid and the distance between nodes is set, so that the degree of the central node is $d = 4$ with $N_{size} = 9$, or $d = 6$ with $N_{size} = 16$ and 25.
- *Ellipse*: for every node the degree is $d = 2$, i.e. each node has a PDR of 100% only with the left and right one-hop neighbours, and the multi-hop path from a sensor node to the sink has a maximum depth $maxD = \lfloor N_{size} \,/\, 2 \rfloor$.
- *Random*: nodes are randomly placed in a square area of $100\,\mathrm{m}^2$. The topology features a portion of the network with high density and other some nodes with a minimum degree $d = 2$.

For each simulated setting, i. e. a possible combination of parameter values, we run 50 independent replications with different seeds, and we report the average value of each metric with its $95\,\%$ confidence interval.

The configuration setting of TSCH and RPL are reported in Table 4.1. For the evaluation, we study the effects of TSCH EB period t_{eb}, TSCH number of channel N_c and RPL minimum interval I_{min} on the following performance metrics:

Tab. 4.1: Setting for TSCH- and RPL-parameters

Parameter		Value
Simulation time		30 min
Simulation runs		50
TSCH slotframe length	L_{mc-SF}	101
TSCH timeslot duration	t_s	10 ms
TSCH scan interval	t_{scan}	1 s
TSCH number of channel	N_c	{4,16}
TSCH KA period	t_{ka}	{12, . . ., 60} s
TSCH EB period	t_{eb}	{2048, 4096, 8192, 16384} ms
RPL minimum interval	I_{min}	{128, 256, . . ., 4096} ms
RPL interval doubling	M	8
RPL redundancy constant	c	5

- *TSCH synchronisation time*, defined as the time between the transmission of the first EB by the Personal Area Network (PAN) coordinator and the first network-wide TSCH synchronisation.
- *6TiSCH network formation time* (or *DODAG formation time*), defined as the time until all nodes are in the RPL logical topology, i.e. in the DODAG. The period between TSCH synchronisation time and 6TiSCH network formation time is not only characterised by the waiting for the first DIO message at the last synchronised node, but also by possible TSCH desynchronisations, so that some nodes may temporarily leave the network.
- *Number of exchanged control frames*, defined as the number of EB, DIO, DIS and Keep-Alive (KA) packet transmitted during the 6TiSCH network formation time.
- *Energy cost* for the network formation process, estimated summing the different charge drawn by the radio component of each node when it is active (e.g. transmission or reception activities) and inactive.

4.3.2 Simulation of 6TiSCH-MC

This first set of simulations were performed to evaluate the performance and the limits of using *minimal schedule* in different topologies. As can be seen in Table 4.1, we do not strictly use default values for the Trickle timer. In particular, the default value $I_{min,RFC} = 8\,\text{ms}$, together with $t_s = 10\,\text{ms}$ and $L_{mc-SF} = 101$, would cause queue drops in the firsts Trickle intervals. Furthermore, advertiser nodes generate EB

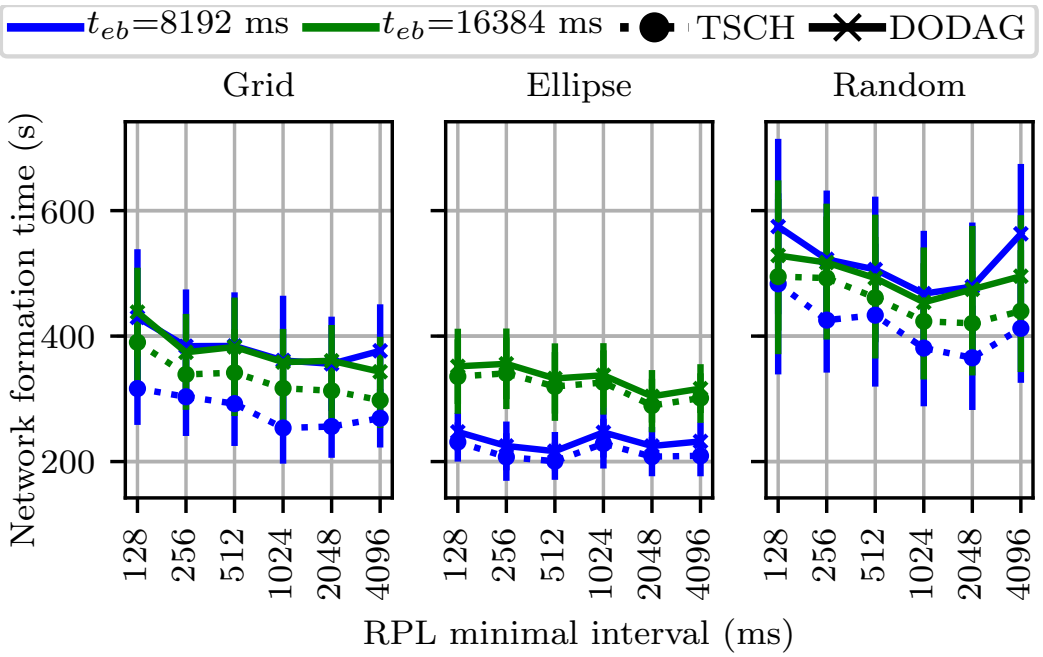
Tab. 4.2: Successful (%) 6TiSCH network formations within 30 min

		Grid			**Ellipse**			**Random**		
		N_{size}			N_{size}			N_{size}		
t_{eb} (ms)		9	16	25	9	16	25	9	16	25
2048	4	0	0	0	70	16	0	0	0	0
	16	0	0	0	2	2	0	0	0	0
4096	4	84	0	0	100	100	86	46	0	0
	16	12	0	0	100	100	26	2	0	0
8192	4	100	20	0	100	100	100	100	2	0
	16	96	0	0	100	98	40	66	0	0
16384	4	100	70	0	100	100	100	100	14	0
	16	98	4	0	94	34	0	92	0	0

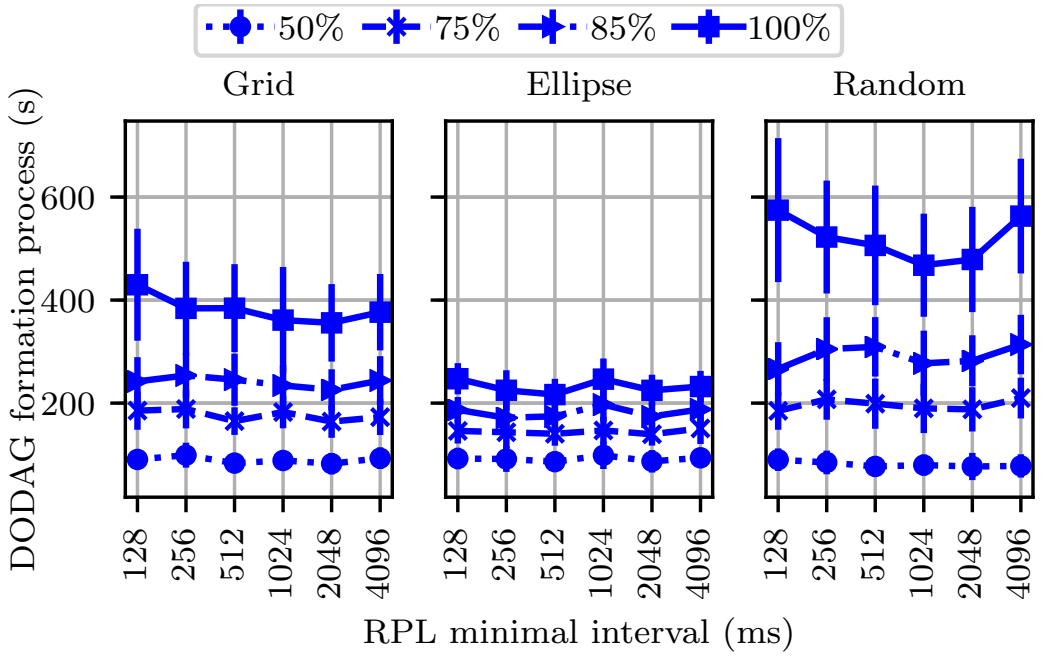
frame with a random EB period in the value ranges between $t_{eb} \cdot [0.75, 1)$. Although the 6TiSCH-MC does not mention this mechanism, open-source implementations of the TSCH (e.g. OpenWSN and Contiki) include it, and [187] shows its advantages in the network synchronisation. In configurations with $N_c = 4$, we assume that the joining nodes are aware of these four available channels, in which transmissions may occur.

Table 4.2 reports on the number of simulations in which the network formation phase is completed within 30 minutes. As can be seen, an improper choice of t_{eb} and the number of channel offset N_c in relation to the network density (i.e. topology) can lead to an unsuccessful network formation, where at least one node is even after 30 minutes not yet operational and cannot transmit any sensed data to the sink. With only one shared slot for broadcasting control frame and $t_{eb} < 4 \cdot L_{mc-SF} \cdot t_s$, the positive effect of applying randomness to the EB periods cannot take place, i.e. advertiser nodes choose with high probability the same slotframe for sending EBs, causing their collision. Besides, also local contentions between EB and other control packets (e.g. DIO, DIS and KA) are critical with this aforementioned setting. These two problems become more and more apparent as the network density increases. Compared to a setting with $N_c = 4$, we observed a linear increase of the network formation time, when $N_c = 16$ is used. For these reasons, we will consider only results obtained with $N_c = 4$ and $N_{size} = 9$ in the reminder of the Section.

In Figure 4.2a the *TSCH synchronisation time* and the *6TiSCH network formation time* are expressed as a function of the RPL minimum interval. As it can be seen, I_{min} shows an influence on the network formation time of random and grid topologies, which is worth investigating. The behaviour is pretty different from the results



(a) TSCH synchronisation time and 6TiSCH network formation time



(b) 6TiSCH network formation time

Fig. 4.2: With $N_c = 4$ and $N_{size} = 9$, we recognised the lowest average DODAG formation time of about 216 s in the ellipse topology. For the grid and the random scenarios the average lowest DODAG formation time are respectively 355 s and 467 s.

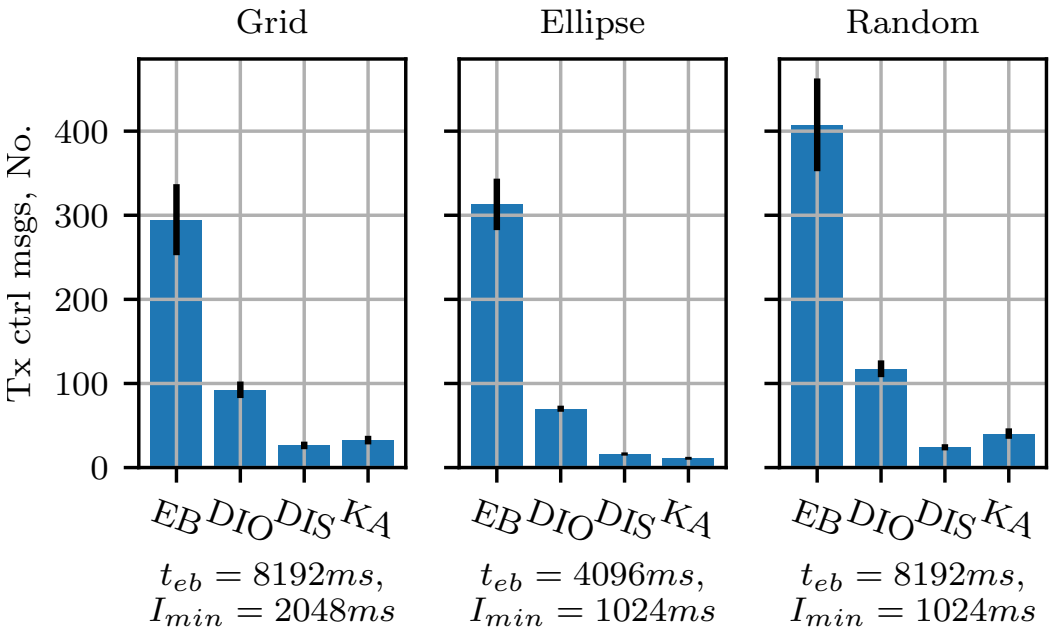


Fig. 4.3: Art and average number of control frames exchanged until all nodes are simultaneously in the DODAG. For each topology ($N_{size} = 9$), we select the best parameter setting concerning the network formation time

presented in [180] for IEEE 802.15.4-2011, where decreasing the RPL minimum interval yields to a reduction of DODAG formation time. One reason is that the configuration of the TSCH slotframe interferes with the Trickle mechanism and implicitly sets a range for the optimal value of I_{min}. Setting the RPL minimum interval to a value significantly shorter than the *minimal slotframe* duration (e.g. $I_{min} = I_{min,RFC} << L_{mc-SF} \cdot t_s$) results in buffering several DIO messages in the transmission queues (or even dropping them due to buffer overflow).We do not appreciate the influence of the RPL minimum interval when we deal with a sparse topology as the ellipse. This fact can be explained considering the limited effect of the Trickle-based timer on this type of network.

Another insight from Figure 4.2a is the significant difference in elapsed time between the network-wide TSCH synchronisation and the DODAG completion, which can be observed in the three topologies. The higher physical density in the random or grid topologies makes collision of EB or KA frames more likely to occur than in the ellipse network, especially when the number of joined nodes increase. Consequently, desynchronisation events are more likely to happen (and they were observed) in such topologies and cause an additional delay of about $100\,\mathrm{s}$, i.e. circa 100 *minimal slotframes*, after the network-wide TSCH synchronisation. Figure 4.2b, which reports on the dynamic of the DODAG formation process in the three scenarios, confirms this

behaviour. The time, until half of all nodes have joined the DODAG, is approximately the same in all scenarios. In the second half of the network formation, the results show how the minimal schedule slows down the completion of the process in dense network.

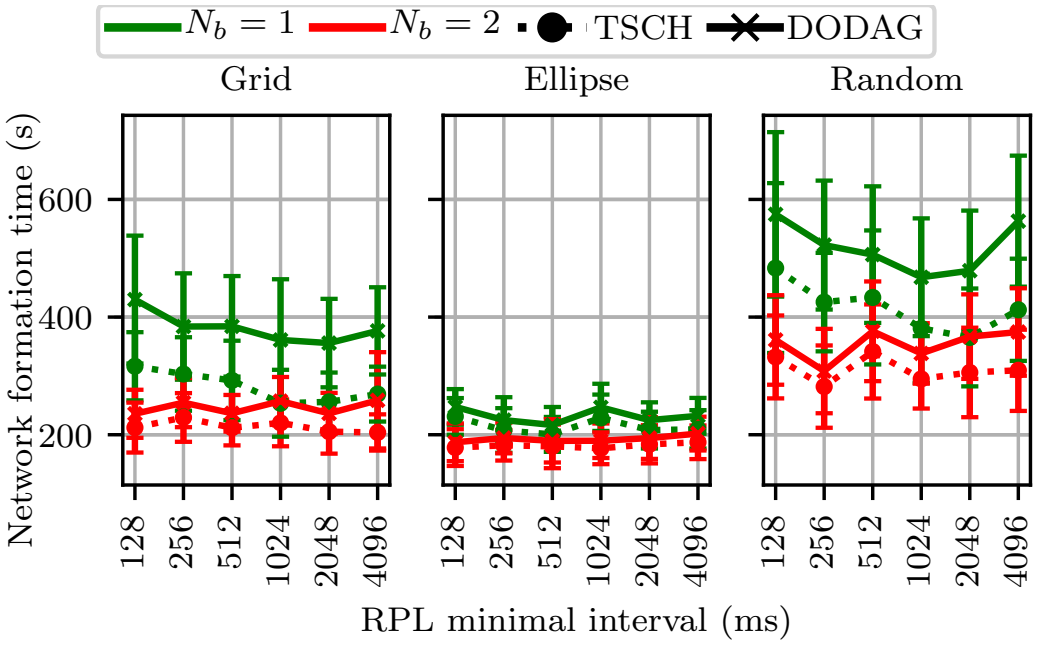
We can see in Figure 4.3 how EBs are the dominant element in all networks. The introduction of an EB transmission strategy, where the EB period is mapped to the Trickle interval, has the potential of reducing the control overhead, but also the drawback of hindering the synchronisation of new nodes [189]. The valuable number of KA in the grid and random scenarios can be explained considering the collisions of frames from neighbouring nodes in these topologies.

4.3.3 Adding Additional Links to 6TiSCH-MC

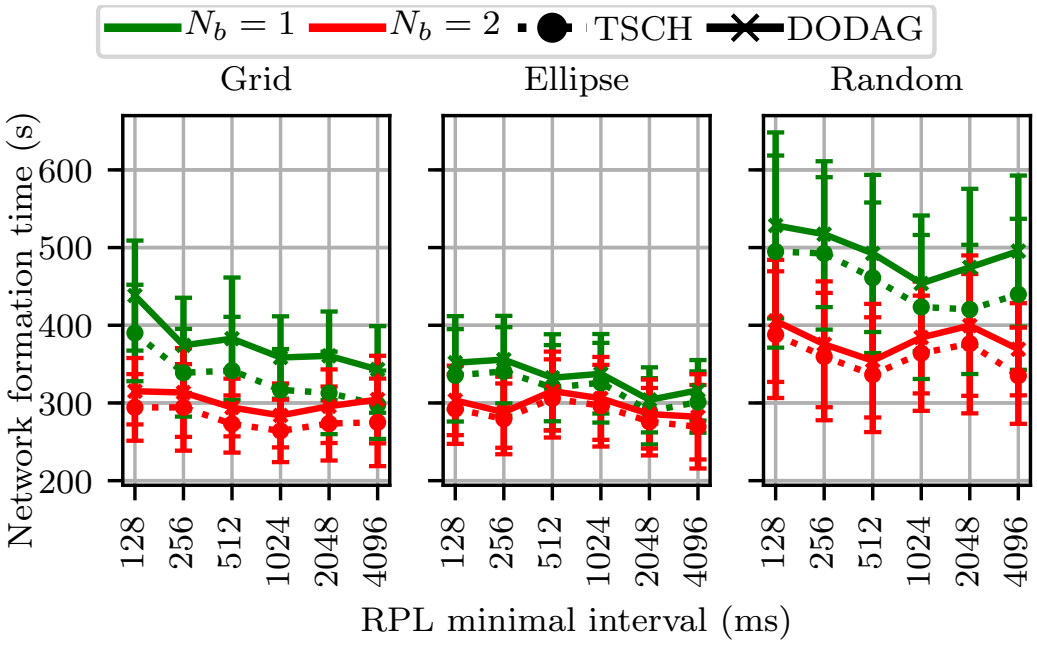
In this experiment, we investigate the influence of an additional slot in the *minimal schedule* on the time and energy consumed for the network formation. We follow the guideline given in [188] allocating $N_{ss} = 2$ shared slots in the basic schedule. Thus, the second slot will be used, if any frame in the transmission queue is present after the first timeslot.The first remark is that the number of configurations affected by unsuccessful network formation decreases. This is an expected behaviour since with a higher value of N_{ss} there is a reduction of local contentions between control packets. Figure 4.4b shows a valuable reduction in the average time spent for the network formation. It is also interesting to observe, how the time gap between TSCH synchronisation and DODAG completion shrinks. The 6TiSCH-MC recommends that EBs are queued with a higher priority than messages coming from higher layers. When only one shared slot is available, the transmission of RPL packets, generated by the Trickle algorithm, may be significantly delayed or dropped, causing the behaviour described in Section 4.3.2. With $N_{ss} = 2$, there is a more efficient messaging and, therefore, a quicker transition from the state *synchronized* to *advertiser* for every joined node.

The drawback of adding links to the *minimal schedule* is a higher base duty-cycle, since every node is active at least in these broadcast slots. For example, setting $N_{ss} = 2$ and $L_{mc-SF} = 101$ results a duty cycle of about $0.2\,\%$. However, the significant reduction of the time spent for network formation causes valuable energy savings, especially in grid and random topologies, as we can see in Figure 4.5a and 4.5b, which present an estimation of the charge consumed for the whole network and for the node with the highest consumption, respectively. This fact is a

consequence of the smaller time spent in waiting for EB by an activated node and of the reduced number of collisions.

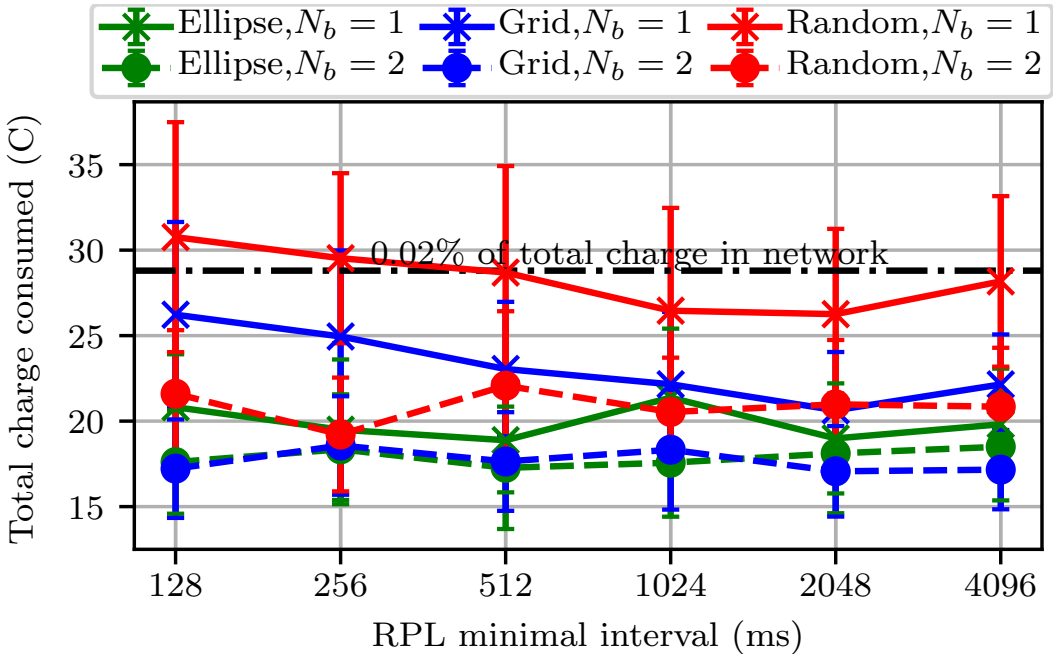


(a) $t_{eb} = 8192\,ms$

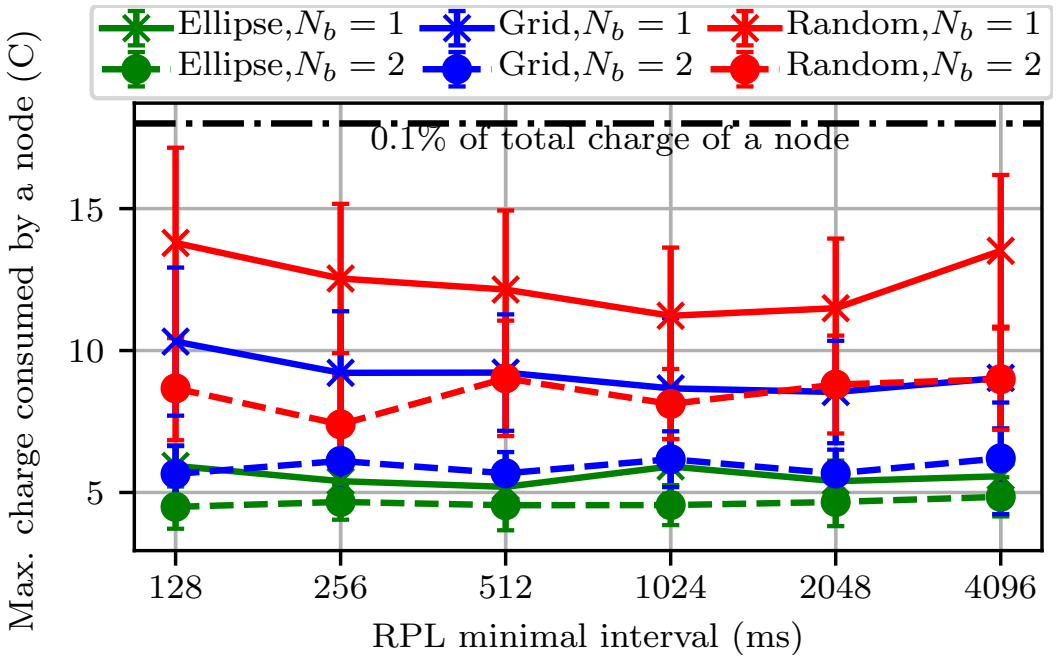


(b) $t_{eb} = 16384\,ms$

Fig. 4.4: Time saving for the network formation process by allocating $N_{ss} = 2$ shared slots for the transmission of EBs and RPL messages with $N_c = 4$, $N_{size} = 9$ and $t_{eb} = 8192\,ms$ (a) or $t_{eb} = 16384\,ms$ (b)



(a) Total charge consumption



(b) [Highest charge consumption in a node

Fig. 4.5: We assume powering every node (except the sink) with a 2xAA battery pack, which carries $18\,000\,\mathrm{C}$ ($2 \times 2500\mathrm{mA\,h}$) and we select as parameter set $N_{size} = 9$, $t_{eb} = 8192\,\mathrm{ms}$, $N_c = 4$. We use as current consumption the values inferred from the datasheet of the TI CC2538 [231], a wireless microcontroller System-on-Chip commonly used for IEEE 802.15.4 applications.

4.3.4 Recommendations

This simulation study allows us to derive a set of guidelines on how to choose the TSCH and RPL parameters for a given topology, as reported in Table 4.3. Foremost, we recommend implementers of 6TiSCH networks to set the I_{min} parameter slightly over the minimal slotframe duration, while the t_{eb} should be set to a value that enables the positive effect of applying randomness in selecting the generation instant of EB messages, e.g. $t_{eb} \geq 4 \cdot L_{mc-SF} \cdot t_s$.

In addition, in dense topologies, the number of shared slots used for broadcasting is a crucial factor, and the use of at least $N_{ss} = 2$ is recommended. The duplicated duty-cycle is compensated by a reduced average joining time of each single node, and by a more likely successful delivery of frame in the whole network.

The rationales of the two additional recommendations listed in Table 4.3 are the following. First, selecting the length of the sloftframe L_{mc-SF} coprime to the number of channel offset N_c enables the TSCH channel hopping mechanism to use regularly all the available frequencies (c.f. Section 2.4.3 and [109]). Secondly, we can easily reduce the 6TiSCH network formation time by restricting the channel hopping sequence, i.e. decreasing N_c from 16 to 4. As reported, we observed this straightforward fact in our experiments. In indoor deployments, a good candidate set for the channel hopping sequence contains the four frequencies that generally experience less WiFi interference, i.e. channels 15, 20, 25, 26, as shown in [46].

Tab. 4.3: Recommended Settings

Parameter		6TiSCH-MC specifications	Our recommendations
RPL minimum interval	I_{min}	8 ms	$\gtrsim L_{mc-SF} \cdot t_s$
TSCH EB period	t_{eb}	Tunable	$\geq 4 \cdot L_{mc-SF} \cdot t_s$
TSCH number shared cell	N_{ss}	1	≥ 2
TSCH slotframe length	L_{mc-SF}	Tunable	Coprime with N_c
TSCH number of channel	N_c	16	4

4.4 Enhancing the 6TiSCH-MC with MACA

Our performance evaluation has shown that allocating an additional shared cell for the transmission of EBs and RPL messages helps in speeding up the network formation in dense topologies. Besides, using this approach, the joining nodes spend less time listening for the first EB, which leads to less energy consumption

during the network formation phase, if compared to the energy depleted by 6TiSCH-MC for this process. This section investigates how to further increase the number of transmission opportunities for bootstrapping traffic in the *minimal slotframe*. Specifically, we introduce the *MACA*. It coordinates the scheduling of $N_{adv} \geq 2$ active cells among neighbouring nodes in a dynamic and distributed way. This mechanism aims to reduce the collisions of control messages and improves the 6TiSCH network formation in terms of time required by nodes to join the network.

4.4.1 Design Principles of MACA

We present below the principal elements of the MACA algorithm:

Advertisement cells with options and states: Using MACA, the matrix-like TSCH schedule of a node contains at least $N_{adv} \geq 2$ advertisement cells. Only one of these cells has the same link option as the shared slot recommended by the 6TiSCH-MC. That is, a node selects merely one cell per slotframe for broadcasting EBs and DIO messages and schedules the other advertisement cells only for receiving the control messages of neighbouring nodes. Locally, a node maintains four different states for each advertisement cell: *idle*, *dedicated*, *shared*, and *congested*. A cell is idle when no control message was received. It is dedicated, if only one neighbouring node is using it for broadcasting. When more nodes in the neighbourhood transmit on the same cell, MACA considers the cell as congested if the number of collisions is greater than γ_c; otherwise, the cell is considered as shared. Two bits per cell are sufficient to encode the node's information on its neighbourhood and are included in every control message (both EB and DIO), as explained later.

Position of advertisement cells in the slotframe: For the location of the advertisement cell, previous works suggest two strategies: (i) cells can be consecutive and located at the beginning of each slotframe [177], [228] or (ii) regularly spaced within the slotframe at specific channel offset [109], [174], [176]. Figure 4.6 depicts these two alternative ways of scheduling $N_{ss} = 4$ shared slots in a minimal slotframe of length $L_{mc-SF} = 17$. In designing MACA, we opted for the latter approach, mainly for two reasons. First, spreading the active cells within the slotframe reduces node joining times, as already mentioned by De Guglielmo et al. [109]. Second, this strategy provides a TSCH schedule closely comparable [1] with the one recommended by 6TiSCH-MC,

[1] in term of the node's duty cycle. Nevertheless, the set of channel offset adopted by MACA is not limited to one value, i.e. $co = 0$, as in 6TiSCH-MC

when the size of the *minimal slotframe* is divided by the number of active cells. The MACA algorithm derives the channel offset of the advertisement cells j as follows:

$$co_{adv} = (co_0 - j \cdot \Delta ts) \mod N_c, \tag{4.1}$$

where co_0 is the channel offset of the first active cell, Δts is the distance, in terms of timeslots, between succesive active cells and N_c is the total number of available channels to hop. The sink node sets co_0, e.g. $co_0 = 0$ as in the 6TiSCH-MC, and $\Delta ts = \left\lfloor \frac{L_{mc-SF}}{N^M_{adv}} \right\rfloor$. The equation 4.1 maps any advertisement cell to the same physical frequency f_k on a given slotframe k. However, the frequency f_k changes in the successive slotframes due to the channel hopping mechanism of TSCH (recall Section 2.4.3 and Equation 2.1). In other words, MACA forces the transmission/reception of all control messages on a specific physical frequency during a given slotframe, so that there is only a small variation of the link quality for each advertisement cell. In addition, adopting only a frequency per slotframe lets a node determine the location of each advertisement cell just using the slotframe size L_{mc-SF}, the number of active cells N_{adv} and the specific frequency used for receiving the control message. This simple design choice significantly reduces the size of the MACA IE (see below) and, therefore, the probability of fragmentation of control messages. Specifically, it is no longer required to append the list of active cells, as the IEEE 802.15.4-2015 standards would require for this case (see *TSCH Slotframe and Link IE* in [87]).

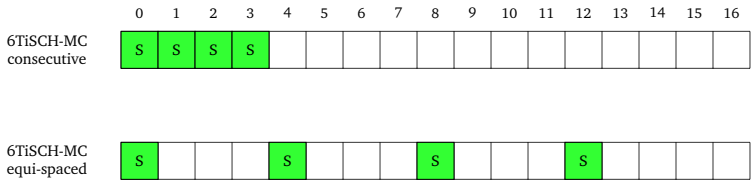


Fig. 4.6: Two simple strategies for scheduling $N_{ss} = 4$ shared cells in a *minimal slotframe* of length $L_{mc-SF} = 17$

Feedback-based reservation of advertisement cells: The MACA procedure's primary goal is to significantly reduce control message collision through a distributed reservation of advertisement cells. When a node joins the network, it contends for one of these cells. To this end, no explicit request-notify messages between nodes are used. Instead, a new advertiser node randomly selects an advertisement cell from the set of empty ones and directly broadcasts EBs and DIO messages on it. Then, the node gathers feedback on this choice from the MACA IEs (see later) included in the control messages received on the other advertisement cells. Specifically, a node continues using the same advertise-

ment cells if all the neighbouring nodes mark it either idle, dedicated or shared. Indeed, the cell's states dedicated and shared suggest a correct reception of control messages; also, the state idle may indicate merely an asymmetrical link to the neighbour node that advertises it. In contrast, when the node receives a control message signalling the selected cell as congested from at least one neighbour, it recognises the reservation as unsuccessful. Thus, it picks another advertisements cell (with probability γ), to resolve the conflict. In doing that, the node considers the set of idle cells in its 2-hop neighbourhood, i.e. cells that both the node itself and all its neighbour evaluate as unutilised. At this place, it is worth underlining that the MACA procedure does not strive for convergence to a conflicting-free allocation of advertisement cells among nodes in the network, as in the related works [168], [176], [177]. Instead, our proposal accepts that nodes can share some advertising cells in a slotframe, mainly for two reasons. First, in dense 6TiSCH network, conflict-free scheduling among node demands allocating a relatively large (and fixed) number of active advertisement cells N_{adv} in the minimal slotframe. This prerequisite may excessively limit the set of TSCH links for data transmissions and, at the same time, it produces a too high average duty-cycle to make a node last for a long time on a battery. Secondly, a collision of control messages in a 6TiSCH network does not inevitably happen when two (or more) neighbouring nodes select the same advertisement cell. Indeed, nodes use this shared advertisement cell, only when they have either EB or DIO messages buffered for transmission at a given slotframe. Considering a typical TSCH period $t_{eb} > L_{mc-SF} \cdot t_s$ and the dynamics of the RPL DIO's transmission interval (regulated by the Trickle algorithm), letting nodes share some advertising cells in a slotframe is a design choice that allows the MACA procedure to find a good compromise between timing and energy performance, as our result show [2].

Dynamic allocation of advertisement cells: The number of advertisement cells N_{adv} is not static and equal for all node of the network as proposed by [109], [176], [177]. Instead, each node sets it considering both the N_{adv} advertised by its RPL parent and the observed local contention in the neighbourhood. Specifically, the sink node starts the network formation by advertising $N_{adv} = N_{adv}^{init}$ active cells, one of which is dedicated and utilised to transmit its own control messages. As soon as a node joins the network, it sets N_{adv} as the

[2] The probability of a collision of control messages in a given slotframe can be expressed analytically as a function of the number of node in the neighbourhood for given TSCH and RPL parameters by following the model proposed in [227]

value advertised by its RPL preferred parent. Then, every τ_{MACA} slotframes, a node updates N_{adv} for the next cycle as follows,

$$N_{adv} = \begin{cases} \min\{2 \cdot N_{adv}, N^{M}_{adv}\} & if\ N^{i}_{adv} \leq \gamma_e \\ N^{pp}_{adv} & otherwise \end{cases}. \tag{4.2}$$

With Equation 4.2, the current local N_{adv} is doubled as soon as the number of idle cells N^{i}_{adv} in the node's neighbourhood is lower than γ_e. Thus, MACA allocates additional empty cells for other potential upcoming joining nodes. In parallel, it helps preventing a conflicting selection of advertisement cells or resolve it on already congested ones. Further, the equation establishes an upper-bound N^{M}_{adv} for the number of advertisement cells allocated by a node, i.e. it limits both the bandwidth for control messages and the duty-cycle during the initial network formation phase. Taking inspiration from [227], MACA sets N_{adv} as a power of 2. This design choice facilitates the dynamic allocation of advertisement cells, since updating N_{adv} at the end of a cycle of τ_{MACA} slotframes does not require any relocation of the cells already scheduled in the previous cycle.

MACA IE: Recall that the MACA algorithm does not require any additional control messages. Still, it exploits the standard transmission of EB frames and DIO messages as set by the TSCH and RPL protocol parameters. MACA only includes additional information in these control messages for the feedback-based reservation of advertisement cells. Precisely, an advertiser node adds a specifically designed IE, dubbed MACA IE, in such control messages before broadcasting it. Besides the current and maximum number of advertisement cells (i.e. N_{adv} and N^{M}_{adv}), the MACA IE contains an allocation bitmap. This bitmap includes two bits for each active cell in the minimal slotframe for encoding their four possible states (idle, dedicated, shared and congested).

Algorithm 1 contains the specific actions performed by a joining node with the MACA procedure as pseudocode. Similar to the standard network formation process described in [101], a node joins TSCH and RPL when it receives the first EB (line 3) and the first DIO (line 8), respectively. Only at this point, it becomes an advertiser node, and it can start transmitting control messages (line 11). Every time a node receives an EB or a DIO, it does not only (possibly) discover a new neighbour, i.e. by considering the source of the control messages (line 2); through the information in the MACA IE included in those messages, a node learns the advertisements cells that are instantiated in the neighbourhood (line 13), and it can also infer which cells are occupied by any of its 2-hop neighbours (line 14). Lines 16 - 34 present the steps

Data:
$N_{adv} \leftarrow 0$: nr. of advertisment cells
$N^{pp}_{adv} \leftarrow 0$: nr. of advertisment cells of RPL preferred parent
$N^{i}_{adv} \leftarrow 0$: nr. of advertisment cells in idle state
$[ts_{adv}, co_{adv}]_{self} \leftarrow$ NULL: coordinate of own advertisment cell
$[ts_adv, co_adv]_i \leftarrow$ NULL: coordinate of advertisment cell used by neighbour *i*
Neighbours = [ ]: list of neighbours
AdvCells = [ ]: list of advertisement cells

1 **foreach** *EB or DIO received* **do**
2 **if** *source i* $\notin$ *Neighbours* **then** add node i to Neighbours
3 **if** *first EB received* **then**
4 TSCH Association
5 $N_{adv} \leftarrow N_{adv}$ from MACA IE
6 Add N_{adv} cells with RX option to the schedule
7 **end**
8 **if** *first DIO received* **then**
9 RPL Joining
10 $N^{pp}_{adv} \leftarrow N_{adv}$ from MACA IE
11 Become advertiser node
12 **end**
13 Update Neighbours list with $[ts_{adv}, co_{adv}]_i$
14 Update AdvCells list with bitmap from MACA IE
15 **end**
16 **every** τ_{MACA} *slotframe* **do**
17 **if** *advertiser node* **then**
18 **if** $N^{i}_{adv} \leq \gamma_e$ **then**
19 $N_{adv} \leftarrow \min\{2 \cdot N_{adv}, N^{M}_{adv}\}$
20 **end**
21 Update N^{i}_{adv}
22 Allocate N_{adv} cells with RX option in the schedule
23 **if** $[ts_{adv}, co_{adv}]_{self}$ == *NULL* **then**
24 Randomly select an empty advertisement cell for TX
25 **if** *negative feedback for* $[ts_{adv}, co_{adv}]_{self}$ **then**
26 **if** *rand(1)* $< \gamma$ **then**
27 Continue using $[ts_{adv}, co_{adv}]_{self}$
28 **else**
29 Randomly select an empty advertisement cell for TX
30 **end**
31 Allocate the $[ts_{adv}, co_{adv}]_{self}$ cell with TX option in the schedule
32 Update MACA IE for including it in each EB or DIO
33 **end**
34 **end**

Algorithm 1: Multiple Advertisement Cells Allocation (MACA)

periodically performed for updating the *minimal slotframe*: first, a node checks if the number of idle cells available in the neighbourhood is sufficient for assuring that new joining nodes can use one of them without causing collisions; then, a decision on the own cell for broadcasting is made considering the obtained feedback and the gathered view of the 2-hop neighbourhood; finally the content of the MACA IE is updated and included in each EB or DIO that will be transmitted in the next cycle of τ_{MACA} slotframe.

Figure 4.7 serves as example of the *minimal slotframe* and the corresponding IE when a MACA algorithm is adopted. The node has allocated only two of the potentially four advertisement cells. The first one, located at $ts_{adv} = 0$, is idle (00 in the allocation bitmap). The second one, located at $ts_{adv} = 8$, is dedicated, i.e. used to transmit EBs and DIO messages (01 in the bitmap).

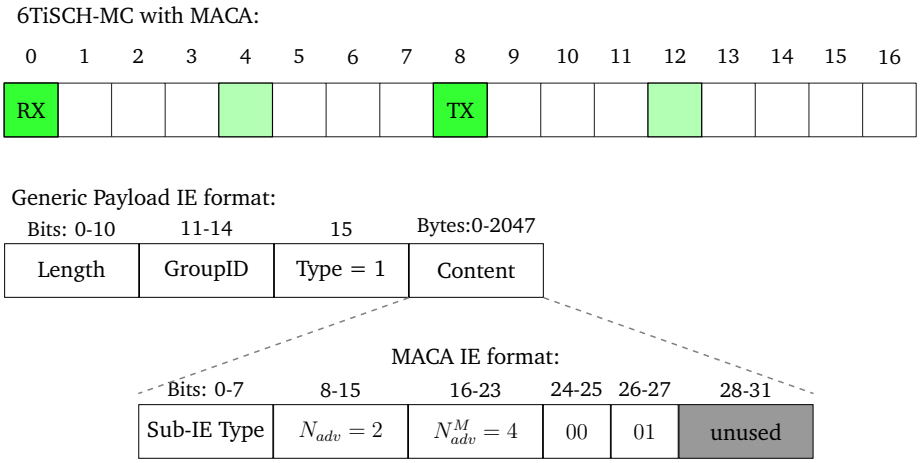


Fig. 4.7: Example of MACA cell allocation and corresponding sub-IE content with $N_{adv} = 2$, $N^M_{adv} = 4$ and a *minimal slotframe* of length $L_{mc-SF} = 17$

4.4.2 Performance Evaluation

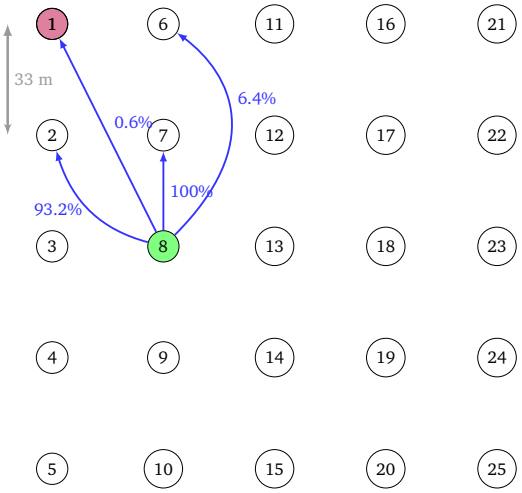
We implemented the MACA procedure presented in Section 4.4.1 in Contiki-NG, the active fork of Contiki. Contiki-NG features an up-to-date and mature implementation of the 6TiSCH protocol stack that 6TiSCH-WG regularly use in its interoperability events. Then, we exploited the Cooja simulator, part of the Contiki-NG suite, to assess the performance of MACA. In the following, we do not limit us to highlight the benefits of reserving additional bandwidth for control messages, as the MACA procedure does. Still, we compare our proposal's performance with other two basic strategies proposed in related works.

Methodology

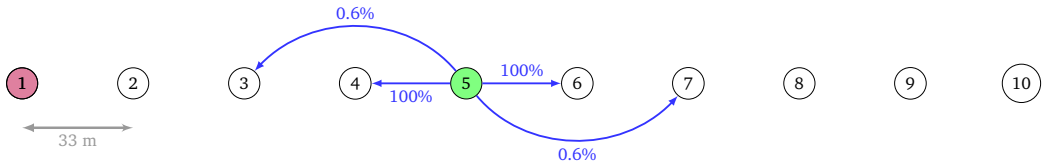
Compared to the simulation methodology presented and adopted in Section 4.3.2, we make some modifications. First, we chose the Multipath Ray-tracer Medium (MRM) as the wireless propagation model in Cooja. MRM is more realistic than the Unit Disk Graph Model (UDGM) used in Section 4.3.2. Indeed, MRM reproduces channels with propagation effects such as multi-path, refraction, diffraction, etc. We set the MRM channel parameters to obtain links with a Packet Delivery Probability (PDP) of $100\,\%$ for nodes that are distant $33\,\mathrm{m}$ from each other. This configuration is close to the reality in case nodes are the current Wireless Sensor Network (WSN) platforms presented in Section 2.2.1. Secondly, we consider here only the two scenarios depicted in Figures 4.8a and 4.8b. The first scenario is a network with 25 nodes arranged in a regular 5x5 grid topology (*grid-5x5*). The second simulated network is a simple chain topology composed of 10 nodes (*chain-10*). The distance between neighbouring nodes leads to an associated average PDP of the links in the networks as reported in both figures.We restrict our experiments to these two scenarios since they have similarities in terms of average node's degree and node's depth with the largest grid and the medium ellipse topologies adopted in the previous Section 4.3.2. Hence, they ensure an analysis of the MACA's performance with different network densities and depths. Finally, we select a fixed configuration of the RPL protocol parameter, without varying the Trickle parameters but setting them as reported in Table 4.4, i.e. following our guidelines previously suggested.

We have already established in Section 4.3.3 that reserving $N_{ss} \geq 2$ shared cells in the 6TiSCH-MC is crucial for a successful and energy-efficient network formation phase. Therefore, in the following, we study the performance of three diverse strategies for scheduling the N_{ss} active cells in a *minimal slotframe* with length L_{mc-SF}. Namely, we compare our MACA proposal with two straightforward extensions of the 6TiSCH-MC. The *6TiSCH-MC consecutive* reserves N_{ss} consecutive shared TSCH-links at the begin of each sloframe, while the *6TiSCH-MC equi-spaced* allocates them equally spaced within the slotframe. The former matches our (preliminary) recommendation given in Section 4.3.3. Instead, the latter is closely comparable to the allocation strategies proposed by [109], [227].

To determine the most suitable allocation approach, we mainly study how long the 6TiSCH network formation process will take and how much energy nodes consume for this process. We vary the TSCH EB period t_{eb} and the number N_{ss} of active cells in the minimal slotframe. Since our MACA procedure allocates the active cells dynamically, we set $N_{ss} = N_{adv}^{M}$ for a fair comparison between the different strategies. Further, we set the other MACA parameters described in the previous



(a) Topology *grid-5x5*



(b) Topology *chain-10*

Fig. 4.8: Network topologies used in Cooja. Node 1 is the sink.

Section as in Table 4.4 after a set of preliminary experiments. The results presented below are averaged over 20 independent replications of each simulated setting. Further, we plot the 95th confidence intervals.

Simulation Results

Table 4.5 reports the number of case in which our MACA algorithm leads to a successful 6TiSCH network formation before the end of the simulation, i.e. within 10 minutes for the 5x5 grid topology and 20 minutes for the chain topology. These results reinforce the recommendation given in Section 4.3.3 about the benefit of adding additional links to the 6TiSCH-MC. Indeed, a rough comparison between Table 4.2 and Table 4.5 suggests how our MACA allocation strategy improves the reliability of the 6TiSCH network formation phase, i.e. all nodes in the network receive at least an EB frame and a DIO message withing few minutes in almost all settings. Precisely, Table 4.6 summarises the **average 6TiSCH network formation time** for successful cases. As can be seen, the network becomes quickly operational when the maximum number of cells allocated for control traffic N_{adv}^{M} is doubled first to two and then to four. This can be easily understood, if we consider that a larger

Tab. 4.4: Simulation settings

Parameter		Value
Simulation time		10 or 20 min
Simulation runs		20
TSCH slotframe length	L_{mc-SF}	101
TSCH timeslot duration	t_s	10 ms
TSCH number of channel	N_c	16
TSCH EB period	t_{eb}	{2, 4, 8} s
TSCH number shared cell	N_{ss}	{2, 4, 8, 16}
RPL minimum interval	I_{min}	128 ms
RPL interval doubling	M	12
MACA number of initial adv cells	N_{adv}^{init}	2
MACA max number of adv cells	N_{adv}^{M}	N_{ss}
MACA idle threshold	γ_e	1
MACA congestion treshold	γ_c	2
MACA cycle	τ_{MACA}	{1, 2, 4}

N_{adv}^{M} implies a lower likelihood of collision when many devices become advertiser nodes and they start broadcasting both EBs and DIO messages. However, the results also show that a further increase of N_{adv}^{M} does not provide an appreciable reduction of the 6TiSCH network formation time. We explain this fact below, where we closely investigate the three different strategies for allocating cells for control messages in the *minimal slotframe*, i.e. *6TiSCH-MC consecutive*, *6TiSCH-MC equi-spaced* and *6TiSCH-MC with MACA* in the two simulated scenarios.

Tab. 4.5: Successful (%) DODAG formations until end of simulations

	grid-5x5				*chain-10*			
	N_{adv}^{M}				N_{adv}^{M}			
t_{eb}	2	4	8	16	2	4	8	16
2 s	100	95	100	100	–	100	100	100
4 s	100	100	100	100	–	75	75	75
8 s	90	95	100	100	–	–	–	–

We first discuss the results obtained with the **topology *grid-5x5***. This scenario represents dense deployments, since a device has (on average) 5.76 neighbouring nodes, from which it should receive control messages. Hence, increasing the number of active cells is crucial in this scenario for obtaining a 6TiSCH network formation within a reasonable amount of time. We expected that our proposal could outperform both *6TiSCH-MC consecutive* and *6TiSCH-MC equi-spaced* in terms of **6TiSCH network formation time**, considering that the MACA procedure aims at a drastic reduction of collisions through a distributed reservation of the active cells

Tab. 4.6: 95^{th} Percentile of network formation time

	N_{adv}^{M}	*grid-5x5*	*chain-10*
$t_{eb} = 2\,s$	2	238.6±31.26	–
	4	145.2±13.48	380.6±48.59
	8	132.8±16.12	377.9±46.79
	16	141.5±21.97	377.9±46.79
$t_{eb} = 4\,s$	2	262.7±32.19	–
	4	214.0±24.60	780.6 ±117.07
	8	210.8±27.17	780.6 ±117.07
	16	215.1±20.38	780.6 ±117.07
$t_{eb} = 8\,s$	2	370.4±61.12	–
	4	381.6±48.96	–
	8	351.4±38.82	–
	16	339.3±32.02	–

among neighbouring nodes. However, the simulation results do not confirm our expectations. Figure 4.9 shows the CDF of the 6TiSCH joining time experienced by nodes for different numbers of active cells N_{ss} allocated in the minimal slotframe. Since the general trend is the same for all the t_{eb} values, we discuss below only the results for $t_{eb} = 4\,\mathrm{s}$. Solid lines refer to *6TiSCH-MC with MACA*, dashed lines correspond to *6TiSCH-MC consecutive*, ant the dotted lines are for *6TiSCH-MC equi-spaced*. Independently from the number of active cells allocated, the *6TiSCH-MC equi-spaced* configuration exhibits the best performance in reducing the overall joining delay of nodes. With this strategy, ca. $70\,\%$ of the nodes (i.e. 17 of 25 devices) are operational after only 2 minutes with $N_{ss} = 4$. The percentage becomes above $80\,\%$ by further increasing N_{ss}. Our MACA strategy is as good as *6TiSCH-MC equi-spaced* only for $N_{ss} = N_{adv}^{M} = 4$ (the figure on the left). Yet, contrary to the other two configurations, we do not observe significant variations in its performance when the number of active cells is further increased to 8 or 16. This behaviour was already prefigured by Table 4.6. To explain this contra-intuitive fact, we need to consider the dynamic allocation of advertisement cells in MACA. In Figure 4.11 we plot hence the relative frequency distribution of N_{adv} set by MACA in each node in the simulations with $N_{adv}^{M} = 16$. Reminding that the MACA algorithm duplicates the current number of active cells only when the neighbouring nodes have reserved all the available ones, the bubble-chart shows that $N_{adv} = N_{adv}^{M}$ cells are allocated only for a small fraction of the time, and even not by all nodes (e.g. not by Node 1, Node 2 and Node 3). Note that to this behaviour contributes the adopted policy of accepting conflicting cells as *shared* when the MACA algorithm does not count collisions on them. Summing up, the MACA strategy tends to be conservative in increasing the

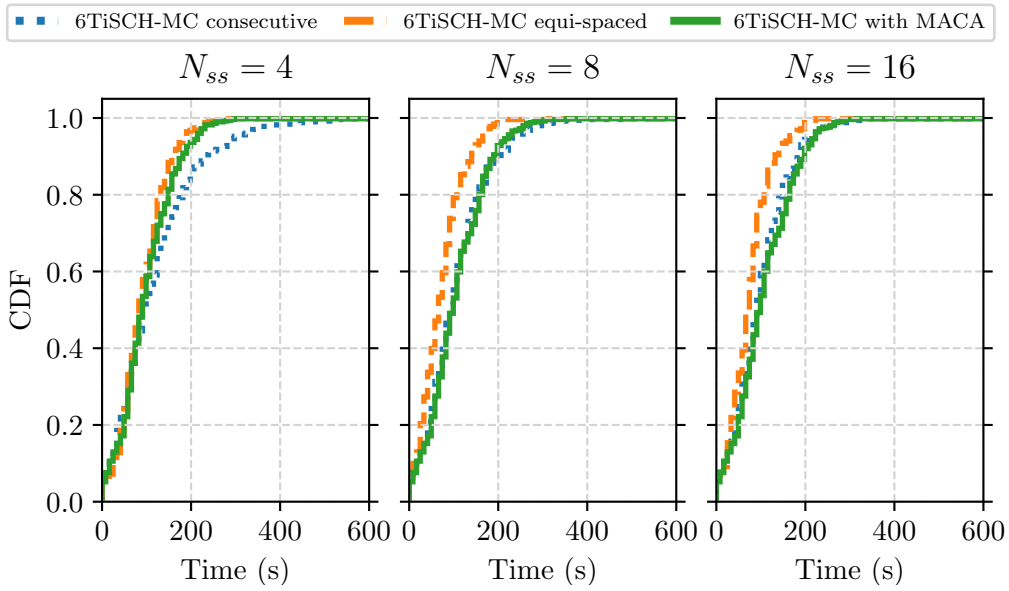


Fig. 4.9: Node's 6TiSCH joining time distribution in the topology *grid-5x5*

active cells (for reasons related to energy-efficiency). Hence, MACA will profit of a greater N_{adv}^{M} only when this value is (anyway) required for a significant reduction of the likelihood of collision between control messages (e.g. N_{adv}^{M} from 2 to 4). Otherwise, a greater N_{adv}^{M} does not lead to an appreciable performance bonus for the 6TiSCH joining time. Nevertheless, we observe in Figure 4.9 an opposite behaviour for the other two configurations, i.e. the time required for joining the 6TiSCH network decreases as N_{ss} increases with *6TiSCH-MC consecutive* or *equi-spaced*. These results are attributable to the several transmission opportunities per slotframe rather than to a significant reduction of collisions in the active cells. Indeed, EBs and DIO messages are generated at random instants in their periodic intervals, buffered in a transmission queue and transmitted in the next transmission opportunity after their generation. Therefore, increasing the number of active cells per slotframe reduces the time these control messages wait in the transmission queue (i.e. queuing delay). The *6TiSCH-MC equi-spaced* gets the most out of this fact as the active cells are spread among the entire minimal slotframe.

Figure 4.10 show the **average number of control messages received by a node** in the *grid-5x5* topology for all the three allocation strategies. Both the number of EBs and DIO messages are reported. As can be seen, allocating active cells equally spaced within the slotframe is crucial for increasing the number of received messages, when nodes can freely choose which cell to use for broadcasting. For every N_{ss} value, the *6TiSCH-MC equi-spaced* strategy outperforms the *6TiSCH-MC consecutive* one. This corroborates the observation in [109]. Intuitively, scheduling all the active cells at

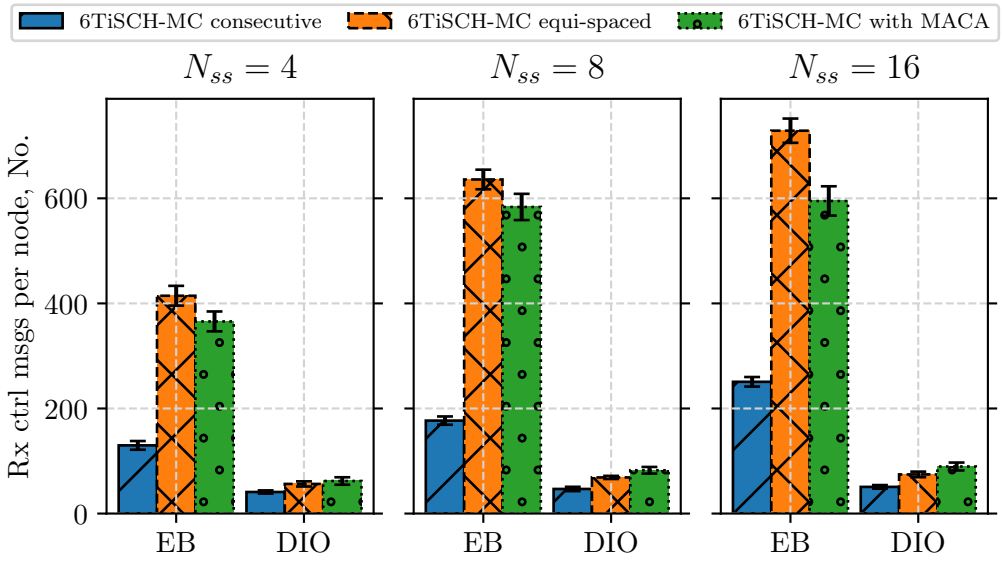


Fig. 4.10: Average number of control message received in the topology *grid-5x5*

the beginning of the slotframe leads to a collision of the control messages generated in neighbouring nodes during the inactive portion of the slotframe. Nevertheless, the distributed cell reservation proposed with MACA also leads to a reliable delivery of control messages during the 6TiSCH network formation, most notably if compared with the performance of *6TiSCH-MC consecutive*. Still, in Figure 4.10, we note a (relatively small) performance penalty of our proposal against the *6TiSCH-MC equi-spaced* configuration in receiving EB frames. An explanation for this observation is the following: *6TiSCH-MC equi-spaced* achieves the shortest 6TiSCH joining time and, therefore, each node has more (simulation) time to generate and transmit additional EBs. It may be worthwhile stressing that the number of transmission opportunities per slotframe, i.e. N_{ss} for *6TiSCH-MC equi-spaced* and only 1 for *6TiSCH-MC with MACA*, is not a relevant factor here, since $t_{eb} > t_s \cdot L_{mc-SF}$ and a node does not generate more than one EB per slotframe. A bonus for our MACA algorithm emerges by considering the DIO messages received in Figure 4.10: it always exhibits the highest reliability in delivery such packets compared to the other solution. However, the performance difference between the considered configurations is less prominent as the one observed for EB frames. That is, the probability that a collision occurs by transmitting a DIO message does not significantly depend on the value of N_{ss} or the allocation strategy. We think the following two motive justify this observation: (i) the Trickle algorithm spreads itself the generation of DIO messages over several minimal slotframes, as the Trickle interval is incremented and its minimum size I_{min} is slightly over the minimal slotframe duration; (ii) the less priority that 6TiSCH-MC gives to DIO messages (when they are buffered)

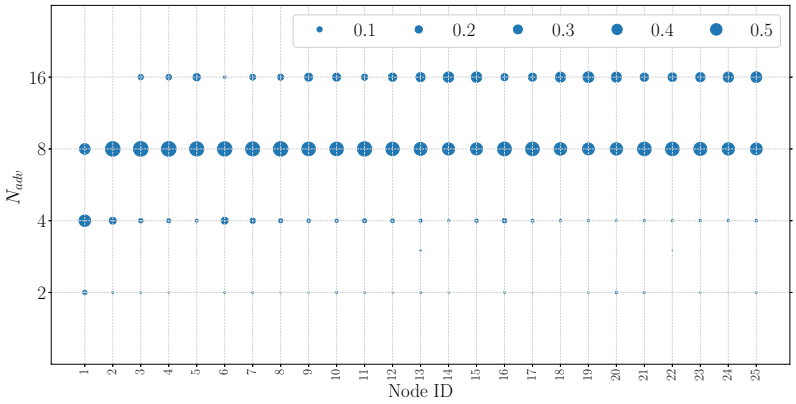
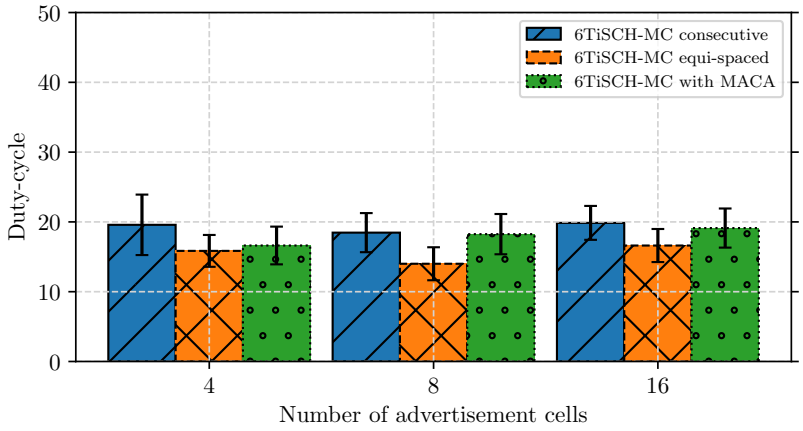


Fig. 4.11: Relative frequency distribution of MACA's N_{adv} for nodes in the topology *grid-5x5*

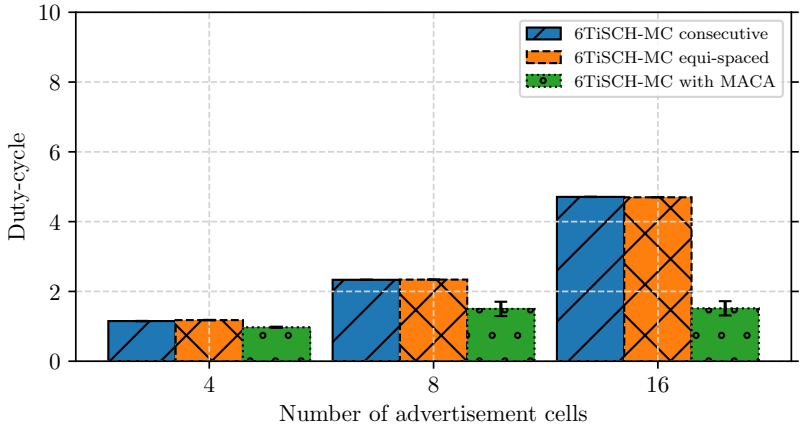
reduces the probability that a node emits a DIO on the first scheduled cells, which is characterised by the most collisions.

We conclude our discussion about the experiments with the topology *grid-5x5* by analysing the **average duty-cycle of nodes** in the network. This metric indirectly evaluates the energy consumption for the 6TiSCH network formation process. Indeed, it expresses the percentage of time in which the nodes remain active for transmission, reception and scan activities during the entire simulation. Figure 4.12a shows that *6TiSCH-MC equi-spaced* exhibits the lowest value in terms of duty cycle (ca. $15\,\%$) for every N_{ss}. At first glance, this is an astonishing result since *6TiSCH-MC equi-spaced* (a) allocates so many N_{ss} active cells per slotframe as *6TiSCH-MC consecutive* and, in doing that, (b) does not consider any mechanism to reduce the energy consumption as our proposed MACA algorithm does. However, the minor overall joining delay resulting from the *6TiSCH-MC equi-spaced* strategy causes valuable energy savings in the network. As already observed in Section 4.3.3, nodes join the TSCH network faster and, therefore, they spent less time waiting for EB, which is the activity that mainly affects the energy-consumption during the network formation. However, a fixed allocation of N_{ss} shared cells becomes a waste of resources (in terms of both energy and bandwidth) as soon as the 6TiSCH network is operational [227]. To estimate the energy spent on network advertising with the three strategies as soon as a node is operational, we plot the sink node's duty-cycle in Figure 4.12b. This device does not listen for EB scanning over different frequency, but it broadcasts both EBs and DIO messages from the first instant of the simulations. As expected, our MACA algorithm exhibits a lower duty-cycle than *6TiSC-MC consecutive* and *6TiSCH-MC equi-spaced*. This fact is due to the dynamic and conservative allocation of active cells provided by our MACA algorithm. As the

value of $N_{ss} = N_{adv}^{M}$ increases, our MACA strategy shows by far its energy-efficiency. For the same reason, we do not observe a steadily increase in the height difference between *6TiSCH-MC equi-spaced* and *6TiSCH-MC with MACA* bars in Figure 4.12a as the N_{ss} value grows, although their difference in terms of 6TiSCH network formation time become larger (cf. Figure 4.9).



(a) Average duty cycle of nodes



(b) Duty cycle of sink node

Fig. 4.12: Duty cycle in the topology *grid-5x5*

Another challenging scenario for the 6TiSCH network formation is the **chain topology**. By placing the sink on the left edge as in Figure 4.8b, nodes become operational in a progressive way from left to right when all devices are switched on at the same time. In this scenario, a small duration of the overall joining process is crucial for avoiding a too excessive battery depletion on nodes with a high depth/distance from the sink. As stated above, most of the energy is wasted for scanning the first

EB. Besides, this operation is especially problematic in a chain topology since each joining node has only one advertiser in its neighbourhood before being operational. Unlike in the dense *grid-5x5* topology, the three allocation strategies do not have to handle a large load on the active cells due to control messages, at least apparently, as we see later.

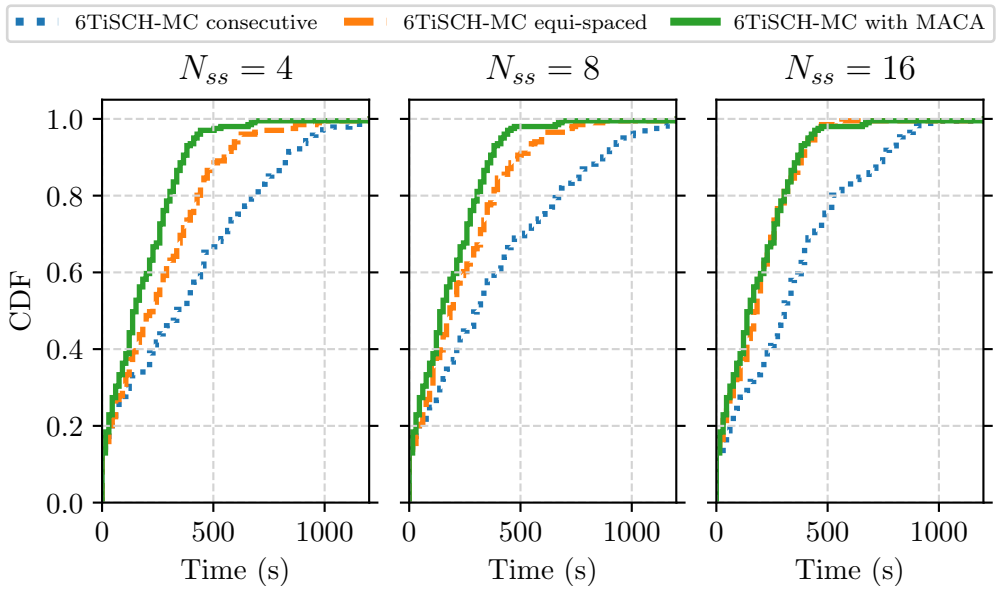


Fig. 4.13: Node's 6TiSCH joining time distribution in the topology *chain-10*

Figure 4.13 show the **CDF of the joining time** experienced by nodes with the three allocation strategies for different $N_{ss} = N_{adv}^{M}$ values. Compared with the results obtained with the *grid-5x5* topology (cf. Figure 4.9), the *6TiSCH-MC with MACA* is now the best candidate for speeding up the 6TiSCH network formation. For the value $N_{ss} = 4$, the delay, until half of all nodes have joined the DODAG, is approximately $75\,\mathrm{s}$ and $190\,\mathrm{s}$ shorter than *6TiSCH-MC equi-spaced* and *6TiSCH-MC consecutive*, respectively. Here, the coordinated reservation of cells provided by our MACA algorithm is decisive. Indeed, it avoids that neighbouring nodes select (with high probability) the same cell at the beginning of the minimal slotframe for broadcasting control messages, as it happens for the other two configurations when the number of active cells is relatively small. Increasing the value of N_{ss} mitigates this problem only for the *6TiSCH-MC equi-spaced* strategy, but it plays a marginal role for the *6TiSCH-MC consecutive*. As we can see, our MACA algorithm does not experience significant variation changing the value N_{ss}. This fact was already prefigured by Table 4.6. It is mostly determined by MACA's ability to allocate only a subset of the maximum number of cells N_{ss}, as confirmed by Figure 4.15, which plots the relative frequency distribution of N_{adv} set by MACA in each node in the *chain-10* topology with $N_{adv}^{M} = 16$.

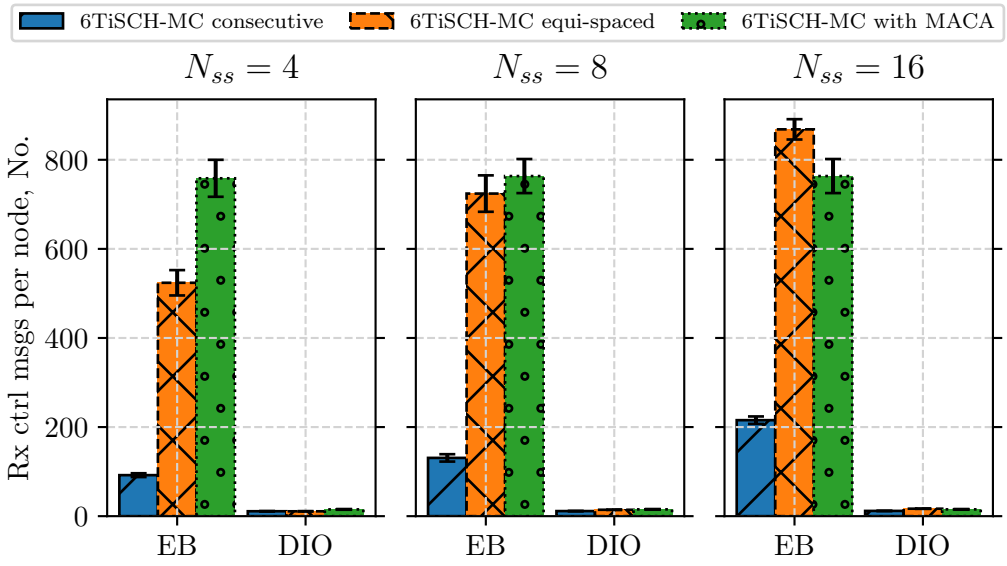


Fig. 4.14: Average number of control message received in the topology *chain-10*

As we have learned from the experiments with the *grid-5x5* topology, the shorter the 6TiSCH network formation phase last in time, the less energy is wasted for this process. Therefore, Figure 4.16a shows that *6TiSCH with MACA* has the lowest value in terms of **duty cycle**, which is not influenced by the parameter $N_{ss} = N_{adv}^{M}$. Figure 4.16b exhibits the duty-cycle of Node 10. This node is placed at the maximum distance from the sink, and, therefore, it is the last node that joins the 6TiSCH network. As we can see, selecting the wrong strategy (namely, *6TiSCH-MC consecutive*) means that this node operates with a duty-cycle larger than 60% during the first 20 minutes. Instead, adopting our MACA procedure limits the worst value of the duty-cycle to $30.5\,\%$. We can give this number a more practical meaning by assuming that the node is a Zolertia RE-Mote [12] powered with a commercial $3.7\,\mathrm{V}$ $600\,\mathrm{mA\,h}$ LiPo rechargeable battery (i.e. $2160\,\mathrm{C}$). In this case, our MACA procedure consumes (approximately) at most $0.4\,\%$ of the node's total charge for the 6TiSCH network formation process.

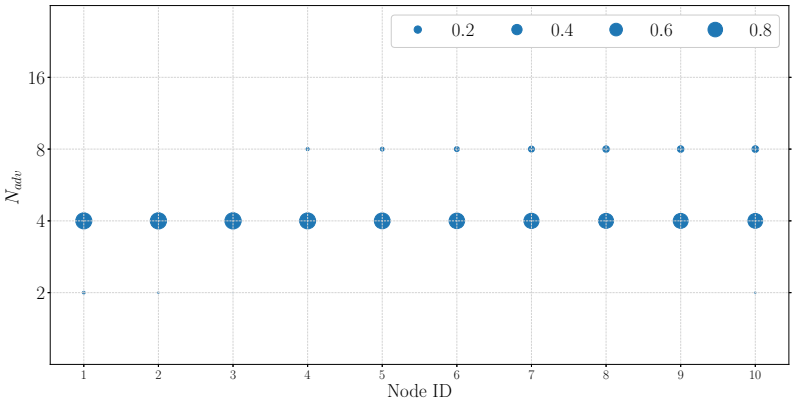
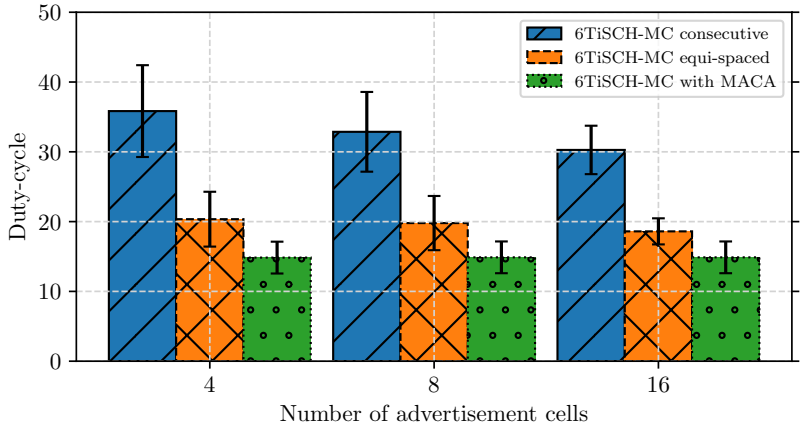
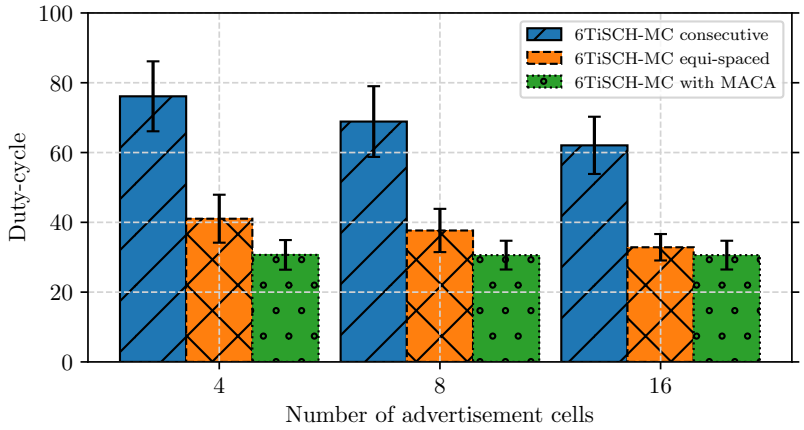


Fig. 4.15: Relative frequency distribution of MACA's N_{adv} for nodes in the topology *chain-10*



(a) Average duty cycle of nodes



(b) Duty cycle of sink node

Fig. 4.16: Duty cycle in the topology *chain-10*

4.4.3 Discussion: To Use or Not to Use MACA

Recall from Section 4.3.2 that the default configuration of 6TiSCH-MC may lead to poor performance. In dense networks, allocating more active cells in the *minimal slotframe* to transmit bootstrapping messages is a central guideline for a reliable and fast 6TiSCH network formation. Our simulative study has demonstrated a significant reduction of local contentions between control messages and a more rapid diffusion of DIO messages. Further, a fast joining time in each node compensates the theoretical higher duty cycle, i.e. the ratio between the number of active cells and the overall slotframe length.

Our MACA procedure achieves all these benefits because the *minimal slotframe* will contain at least $N_{adv} = 2$ advertisement cells. Moreover, no extra control message is generated with MACA. Only few additional bytes are included (in the form of MACA IE) in the transmitted EB and DIO packets.

However, the distributed feedback-based reservation procedure, which is a distinguishing feature of MACA, does not boost the 6TiSCH network formation phase in dense networks as expected. In this scenario, a straightforward spreading of several shared cells within the minimal slotframe (i.e. *6TiSCH equi-spaced* strategy) exhibits better performance in terms of overall joining time and energy consumed than our MACA algorithm. Indeed, *6TiSCH-MC equi-spaced* offers several and distributed transmission opportunities during the generation intervals for EBs and DIO messages. This is sufficient to reduce the collisions of such messages, when they are concurrently transmitted by interfering nodes. In particular, the persistent packet conflicts over successive slotframes does not occur. Furthermore, each node has more transmission opportunities per slotframe, and, therefore, EBs and DIO messages stay buffered for a shorter time until their transmission. Conversely, our MACA algorithm reserves only a cell for broadcasting, producing the opposite effect. However, our proposal schedules the advertisement cells in a dynamic manner, i.e. each node adapts the number of active cells considering if its neighbouring nodes have the opportunity to reliably broadcast the control message. Further, each advertiser node gets either a dedicated or a not congested cell for broadcasting whenever possible. The benefits of these distinctive MACA elements are demonstrated by the performance of MACA in the challenging *chain-10* scenario, where it outperforms the other two straightforward extensions of 6TiSCH-MC.

The conclusion we can draw from the previous discussion is: MACA improves the 6TiSCH network formation without complex management overhead or extra energy cost. In particular, the dynamic allocation of advertisement cells in the *minimal*

slotframe and their decentralised reservation among neighbouring nodes are features that support the adoption of MACA for enhancing the 6TiSCH-MC.

4.5 Chapter Summary

This chapter has investigated the bootstrap phase of a 6TiSCH network, i.e. when nodes progressively join the TSCH network and the RPL routing topology. It is a mandatory phase, before nodes may transmit any sensed data, and, therefore, its performance evaluation is particularly interesting from an installation point of view. In this initial phase, network advertising is a fundamental primitive at the MAC and NET layers. Advertiser nodes broadcast control messages (i.e. EBs and DIO packets) containing all the necessary information for the TSCH network-wide synchronisation and RPL DODAG construction. The network becomes fully operational when all nodes have received an EB and a DIO message. To this end, 6TiSCH-WG has standardised the 6TiSCH-MC. This RFC document defines (a) the recommended configurations of the TSCH and RPL, (b) the necessary coordination between these two protocols and (c) a fixed schedule with only a single shared cell for exchanging the control messages at network bootstrap.

Firstly, this chapter has described the dynamic of the 6TiSCH network formation process in its several steps (e.g. scanning for EBs, synchronising, joining a DODAG), as regulated with the 6TiSCH-MC. In particular, we have shown how the TSCH and RPL parameters govern the generation and transmission rate of EB and DIO messages.

Secondly, we have assessed the performance of the 6TiSCH network formation through extensive simulations. The results have shown that a blind adoption of 6TiSCH-MC may delay or prevent nodes from joining the network, particularly in high node density scenarios. Here, the adoption of the default value from the RPL RFC and a wrong choice of the TSCH EB periods cause collisions of control messages, the starvation of DIO packets in the transmission queue of nodes and several TSCH de-synchronisation events when the network is forming. To address these issues, we have given a set of guidelines on tuning TSCH and RPL parameters.

Finally, an extension of the 6TiSCH-MC has been proposed to avoid prolonged or unsuccessful 6TiSCH network formation. This extension allocates dynamically $N_{adv} \geq 2$ active cells, regularly spaced within the minimal slotframe. A feedback-based, distributed reservation mechanism sets in each node a cell for broadcasting and the other ones for receiving messages in the neighbourhood. The proposed

dynamic allocation strategy has been evaluated through Cooja simulations against other straightforward extensions of the 6TiSCH-MC already presented in the literature. In doing that, we have considered two different scenarios that are awkward for network bootstrap and represent real-world use cases. Performance evaluation results have shown that the MACA strategy leads to reliable delivery of control messages in all experiments. Thus, it improves the 6TiSCH network formation process both in terms of node joining time and energy consumption. Notably, the MACA copes better than the other considered strategies in challenging long chain topologies. Here, the coordinated reservation of advertisement cells provided by our proposal is crucial to avoid collisions when neighbouring nodes start broadcasting EB and DIO messages. In addition, the MACA's conservative approach in increasing the number of advertisement cells, which was observed to cause a little performance penalty w.r.t the other strategies in dense deployments, significantly limits the energy waste during the bootstrap phase of chain topologies.

5 Reliable Link Quality Estimation for 6TiSCH Networks

5.1 Chapter Overview and Contributions

This chapter focuses on the Link Quality Estimation (LQE) problem in IPv6 over the TSCH mode of IEEE 802.15.4 (6TiSCH) networks. Many works have proposed different approaches for accurate LQE in Wireless Sensor Networks (WSNs) (recall Section 3.3.2). Among them, a popular strategy, often adopted in practice, consists in sending periodic probing messages to estimate the link quality towards neighbours. However, this mechanism is not designed for 6TiSCH networks and does not include guidelines on how its operations should be performed on time-slotted environments. As we describe later, even the central concept of *link between two nodes* must be revisited when we apply a LQE strategy in 6TiSCH network, since the physical, the link and the network layer of the 6TiSCH protocol suite bring each a different connotation to it. In order to highlight this problem, we run a set of simulations to measure the performance on popular LQE mechanisms proposed for Routing Protocol for Low Power and Lossy Networks (RPL). Our results show that even the state-of-the-art for accurate link measurements in RPL exhibits a non-negligible estimation error in 6TiSCH networks, which does not guarantee a reliable and stable setup of the network (i.e. routing and scheduling). This is mainly due to the default timeslot allocation strategy that reduces the complex concept of link in 6TiSCH to a simple slotted Aloha channel. Thus, a LQE strategy deduces a link metric from measurements collected on this shared channel that has any correlation to the dedicated links that are actually used for data delivery on the entire network. Based on such results, we design Dedicated Broadcast Slotframe for Probing (DBS-P), our proposal for LQE in 6TiSCH. In DBS-P, dedicated cells are allocated for the transmission of probing messages in order to mitigate the disturbance on LQE caused by collisions in shared cells. Further, we define the structure of the probe messages for estimating the Expected Transmission Count (ETX) on 6TiSCH link following its original definition, i.e. handling link asymmetry (recall Section 3.3.1).

Simulation results demonstrate that DBS-P significantly improve LQE accuracy without introducing a significant overhead.

In short, the main contributions provided by this chapter can be summarised as follows:

- A preliminary set of experiments to highlight that well-acknowledged LQE strategies for RPL, but not designed having Time Slotted Channel Hopping (TSCH) in mind, do not perform well in a time slotted environment
- The presentation of DBS-P, our proposal for LQE in 6TiSCH networks, which defines an extension of the 6TiSCH Minimal Configuration (6TiSCH-MC) and allows a collision-free transmission of broadcast probe messages even during the network bootstrap
- A performance evaluation of DBS-P that demonstrates how our proposed strategy assists nodes in gaining an accurate view of their neighbourhood

The remainder of this chapter is structured as follows. We illustrate the different perspectives in defining links in a 6TiSCH-based network in Section 5.2. Then, we analyse the shortcomings of existing LQE strategies for WSN on 6TiSCH networks in Section 5.3. Consequently, we propose an LQE framework that improves the accuracy of the estimation significantly and support the RPL protocol in establishing a functional 6TiSCH network in Section 4.4. Finally, Section 4.5 summarises the chapter.

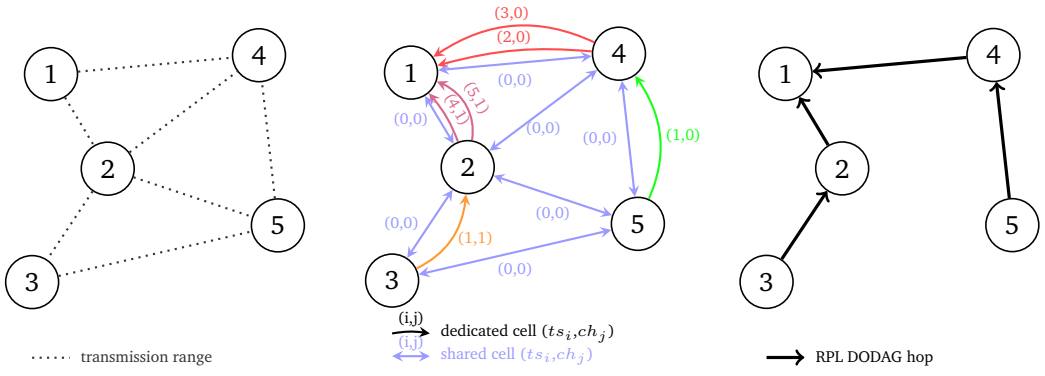
5.2 The Concept of *Link* in 6TiSCH Networks

In this section, we explain how manifold the concept of *link* in 6TiSCH is and how this fact impacts the design of LQE for 6TiSCH networks. To this end, let us consider Figure 5.1 that shows an exemplary 6TiSCH network composed of five devices. We assume that (i) the network formation phase is concluded and (ii) a scheduling algorithm has already allocated the communications. Each subfigure illustrates an aspect of a 6TiSCH link as follows.

The *physical* is the first aspect of a link in a 6TiSCH network. When two nodes are within radio transmission range of each other, a *physical link* between them exists. We express this concept through dotted lines in Figure 5.1a. On a physical link, 6TiSCH exploits frequency diversity, i.e. up to 16 channels are available, and subsequent transmissions occur on different frequencies. Since there is no

correlation across frequency as shown in [42], [69], a physical link has different characteristics from channel to channel, and the Packet Delivery Ratio (PDR, see Section 2.3.1) between two communicating neighbour nodes may exhibit significant variation in different frequencies. Recall from Section 3.3.4 that LQE acquires the denotation of *channel quality estimation (CQE)* in this context. CQE enables whitelisting/blacklisting mechanisms to restrict the communication only on channels where the physical links are known to be of good quality.

The Medium Access Control (MAC) protocol adds in a 6TiSCH network an additional connotation to the concept of link. Indeed, TSCH determines *when* and *how* to use the physical link between neighbours. To this end, a *TSCH link* or *cell* is identified both by the coordinates $[ts, co]$ (i.e. *timeslot* index and *channel offset*) and by its options, i.e. *Tx, Rx* and/or *shared* (recall Section 2.4.3). For instance, in Figure 5.1b, TSCH relates three cells (i.e. the three annotated arrows) to the physical radio link between *node 1* and *node 4*; specifically, this physical link is scheduled twice for a contention-free transmission from *node 4* to *node 1*, but also once with concurrent access regulated by a Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism. In general, the two *dedicated cells* host unicast data traffic, whereas the *shared cell*, which is simultaneously scheduled by each node in the network, offers basic connectivity among all nodes. It means that in 6TiSCH networks various communication patterns take place on the same physical link at different time. The consequence for LQE is, that the link statistics collected on shared cells may be disturbed by collisions. Thus, the resulting link metric offers a rough indication of the congestion on this broadcast channel, and it may exhibit a loose correlation with the packet delivery ratio of that link in dedicated cells. Considering this fact, a proper LQE strategy should differentiate link measurements that were taken at diverse kinds of TSCH-link and aggregate cell



(a) At PHY layer **(b)** At MAC layer **(c)** At NET layer

Fig. 5.1: The manifold concepts of *link* in a 6TiSCH network.

statistics by grouping together measurements of equivalent cells. These cells are represented with different colours in Figure 5.1b, and, for instance, they are cells scheduled at different coordinates but with the same neighbouring node and the same options.

At the network layer, a routing protocol such as RPL selects a subset of high-quality physical links for forwarding packets on, and it ensures that each node has at least one appropriate neighbour to communicate with. As depicted in Figure 5.1c, the routing protocol builds a logical topology on top of the physical network. For RPL, links are child-parent relationships between nodes. RPL takes routing decisions continuously for reacting to changes in the environment (e.g. external interference or node failures), and it relies on the estimation of the quality of the physical and TSCH links, i.e. on a LQE mechanism. On the one hand, misleading routing decisions are possible if the MAC layer aggregates statistical data from all cells a node has instantiated, as above stated. On the other hand, if only the measurement from data packets exchanged on the dedicated cells were used to computes link metric, RPL would have available only the statistics for the established parent-child links and it would have no chance to select a better path proactively.

Recall from Section 3.3.1 that a LQE mechanism consists of three components: (i) link monitoring, (ii) link measurements and (iii) metric evaluation. Thus, designing a LQE mechanism for 6TiSCH network requires to assess for each of these component the specific definition of link, which the 6TiSCH protocol suite introduces.

Moreover, the dynamic allocation of cells among communicating node is a further challenge for designing an LQE mechanism for 6TiSCH. In the initial network phase, the TSCH schedule is mostly composed of shared cells for neighbour discovery and for basic unicast communication with any neighbour. Only successively, nodes allocate the dedicated cells for data communication in the schedule. Therefore, an LQE mechanism should take into account the presence of these two separate phases and adopt proper LQE methods for each of them.

In general, it is beneficial having coarse information about the quality of the channels and the reachability of all the nodes in the neighbourhood in the initial phase to classify their importance and to feed the routing protocol [198], [217]. Successively, accurate estimation of the established parent-child links should take a predominant role in the design of the LQE strategy and support the dynamic allocation of dedicated cells on those links.

Next, we describe how the manifold concepts of link in 6TiSCH networks is responsible for the inadequacy of state-of-the-art LQE mechanisms in such networks. In

particular, their shortcomings concern the initial phase of the network, when LQE is crucial for building the routing topology and the network schedule for a reliable and timed data delivery.

5.3 Shortcomings of Existing LQE Strategies on 6TiSCH Networks

This section investigates how well currently used LQE strategies perform on 6TiSCH networks. We hypothesize that the satisfying results obtained by those strategies with traditional CSMA/CA-based MAC are no longer valid for 6TiSCH and here a non-negligible estimation error may arise, especially during the network bootstrap phase. This fact is due to the default timeslot allocation strategy suggested with the 6TiSCH-MC standard (recall Section 4.2). This schedules only shared cells for transmitting and receiving all the bootstrapping traffic. We demonstrate that collected link measurements are disturbed by the collisions experienced on shared cells and how the inaccurate estimation of the link qualities affects the network performance.

5.3.1 Problem Statement

An accurate and stable LQE is of prime interest in 6TiSCH networks for several reasons. First, the RPL routing protocol employs the measurements offered by LQE to pick the links with the highest quality for building the best routing path between nodes. Here, an accurate ETX of the links as routing metric results in a minimization of the aggregated number of transmissions (and the same for both end-to-end latency and energy consumption) [197]. Second, if accurate ETX values are available, then a 6TiSCH scheduling function can take into account the average number of retransmissions required for data delivery and allocate more cells in a slotframe for links of intermediate or poor quality. As shown in [232], this approach helps to maximize the performance of a 6TiSCH network in terms of end-to-end latency and reliability for the following two reasons: (i) it increases the likelihood that each node successfully delivers data within a TSCH slotframe and (ii) packet drops due to buffer overflow are avoided. Finally, LQE in 6TiSCH should handle the dynamic behaviour of links and filters short-term variation out. That avoids unnecessary network reconfigurations such as preferred parent changes in RPL, which are very energy- and time-consuming as shown in [64], [232]. Indeed, a change in the Destination

Oriented Directed Acyclic Graph (DODAG) topology implies a re-allocation of the reserved cells in the TSCH-matrix, which is not instantaneous and creates (i) an additional overhead of control packets and, potentially, (ii) dropping of some data packets due to buffer overflows.

Section 3.3 have described the different approaches proposed in literature for an accurate and stable LQE in WSNs. We gave special attention to the ETX metric and the mechanisms for its computation when the RPL protocol is adopted. Further, we pointed out that LQE mechanisms proposed for RPL so far assume IEEE 802.15.4-2011 (or other precedent versions) either with CSMA/CA [74], [184], [198], [216] or with ContikiMAC as radio duty cycling mechanism [201], [212], [214]. Thus, we can not expect that they do perform in the same way when the TSCH protocol is adopted at the MAC layer as in 6TiSCH networks.

Rather, we claim that a fallacy in the LQE strategy arises if it deduces ETX values from a set of measurements taken on shared cells and then adopts these values for routing or scheduling decisions, which involves dedicated links or cells. However, that is what happens by applying popular LQE strategies in 6TiSCH networks when the default 6TiSCH-MC is active. In particular, this fallacy concerns the network bootstrap phase, when collected topology information are crucial for computing the optimal routes for multi-hop data delivery, but only a set of shared timeslots is available for transmissions and thus for probing. Therefore, we believe that the ETX values obtained with popular LQE strategies during this initial phase are more a rough indication of the congestion on the shared cells than a reliable metric for routing and scheduling, i.e. for reliable data delivery on the entire network.

5.3.2 Performance Study

To support our claim and highlight the problem of existing LQE strategies in 6TiSCH networks, in the following, we describe the results from a simulative study. We evaluate the performance of the two LQE techniques that are available in the RPL implementation of the Contiki Operative System (OS): a default strategy that passively exploits data traffic and an optional active probe strategy that, instead, periodically transmits specific probe messages to estimate the quality of the wireless links towards neighbours.

With the *default passive strategy*, RPL records statistics on success and failures only for the links used by data traffic, i.e. links that already belong to the DODAG topology. Still, other wireless channels between neighbouring nodes (i.e. idle links)

are ignored. In other words, every node assesses the quality only towards the few neighbours (i.e. the current preferred parent and any former parents) that are ever addressed with unicast data packets. As a result, such an approach produces partial, poor accurate and often outdated link estimates [202], [207], [212].

To overcome those limitations and to provide a more accurate link selection, the implementation of RPL for Contiki also supports the *active link probing mechanism* proposed by Duquennoy et al. in [197]. In particular, when RPL runs this optional strategy, each node periodically sends a unicast DODAG Information Object (DIO) message to one of its neighbours. The target of the probe is selected considering the role of each neighbour (i.e. either current preferred or potential parent) and the freshness of their metric.

We select this active link probing technique in our simulation study since it is the state-of-the-art for accurate link measurements in RPL. Our goal is evaluating how well this technique succeeds in estimating the ETX of links in 6TiSCH networks. In particular, its performance during the network bootstrap phase is crucial. Indeed, during this phase, both the RPL DODAG construction process at routing layer (c.f. Section 4.2) and the scheduling functions (c.f. Section 2.4.5) leverage on link statistics for building a directed tree towards the sink and an accurate TSCH network-wide schedule. It is important that each node estimates the quality of the links towards its neighbours and that these estimations are accurate. Otherwise, suboptimal routing paths and a wrong allocation of transmission opportunities in the schedule are the undesirable consequences.

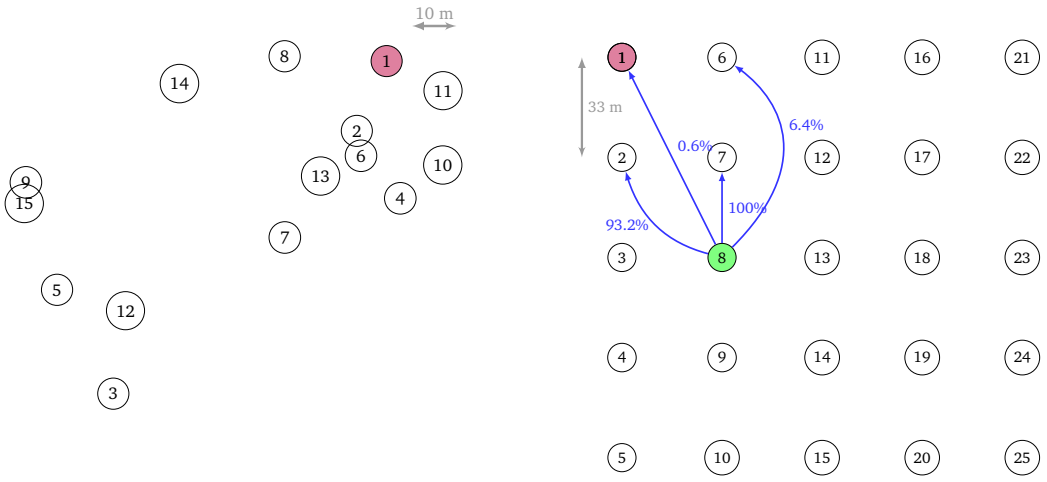
Methodology

As mentioned earlier, our goal is evaluating how well the state-of-the-art strategies for accurate link measurements in RPL succeed in 6TiSCH networks, where the Time Division Multiple Access (TDMA) and multi-channel TSCH technology replaces the traditional CSMA/CA-based MAC protocols. We opt for a simulative investigation, rather than for an experimental campaign. This choice allows us to easily control the characteristics of each link in the network. Specifically, we can set the Packet Delivery Ratio (PDR) of wireless links, and, therefore, the nominal ETX values for each link (e.g. using Equation 3.2 [119] or the definition $ETX = 1/PDR$ [197], [198]. We use these nomimal ETX values as baseline when we evaluate the error in estimating ETX produced by the LQE strategy under study.

We use Cooja, the network simulator included in Contiki OS, and the Multi-path Ray-tracer Medium (MRM) as the radio propagation model. The parameters of the

MRM-model are configured as in Table 5.1 for obtaining radio environments close to the reality in case nodes are the current WSN platforms presented in Section 2.2.1.

We deploy the nodes in the Cooja simulator creating two different topologies, as depicted in Figure 5.2. First, we place them in a 5x5 regular grid (*grid-5x5*), where the sink occupies the upper left corner. This is the same topology adopted in Section 4.4.2. Then, in a second simulation set, we randomly [1] place 15 nodes in a square area of $100\,\mathrm{m}^2$ (*random-15*). Table 5.2 offers metrics characterizing the resulting topologies. The *grid-5x5* scenario consists of 110 links, which are either (very) good or bad (recall the classification of Table 3.1). Moreover, note that the regular grid implies a near uniform distribution of these several links among the nodes. On the contrary, the distribution of the 105 wireless links does not follow a regular pattern in the second simulated scenario, i.e. *random-15*. Here, the nominal neighbour map varies significantly from node to node, so that, for instance, the *very good* links (i.e. $PDR = 1.0$)) are eight at Node-2 and only two at Node-3. Note that the topology *random-15* stands out for the presence of neighbour links in each of the four general categories (i.e. very good, good, intermediate and bad) presented in Section 2.3.1. We select these two different scenarios to highlight that the problem of existing LQE strategies in 6TiSCH is not related to a specific network topology.



(a) Topology *random-15* **(b)** Topology *grid-5x5*

Fig. 5.2: Network topologies used in Cooja. *Node 1* is the sink. Resulting Packet Delivery Ratio (PDR) for *Node 8* are shown in *grid-5x5* topology

Given that we want to study LQE performance in the crucial phase of the network bootstrap, we simulate only the first 15 minutes of a 6TiSCH network, after a simultaneous activation of all the nodes. During this time, nodes only perform the network formation procedure previously described in Chapter 4, and there is no

[1] only the sink node intentionally occupies the upper right corner to dictate a multi-hop topology

Tab. 5.1: Settings of MRM-model in Cooja

Parameter	Value
SNR reception threshold	2 dB
Background noise mean	-100 dBm
Background noise variance	1 dB
Extra system gain mean	-18,45 dB[a] or -20 dB[b]
Extra system gain variance	1,2 dB[a] or 0 dB[b]
Frequency	2,4 GHz
Capture effect preamble	64 μs
Capture effect threshold	3 dB
Default transmitter output power	0 dBm
Receiver sensitivity	-95 dBm
Max path rays	1
Max refractions	1
Max diffractions	0
Refraction coefficient	-3 dB
Reflection coefficient	-5 dB
Diffraction coefficient	-10 dB
Obstacle attenuation	-3 dB/m

[a] in topology *Random-15*
[b] in topology *Grid-5x5*

transmission of data packets towards the sink. In order to avoid an unreliable and long network formation phase that the default configuration of the 6TiSCH-MC would cause in these scenarios (recall Section 4.3.2), we allocate a number of shared slots N_{ss} in the *minimal schedule* greater than one, and we restrict the subset of available frequencies for transmission to four, i.e. $N_c = 4$. Note that the discrepancy from the 6TiSCH-MC, coming from setting these two parameters in the just described way, is beneficial for the LQE strategy in RPL. Indeed, all nodes join the 6TiSCH network quicker and they have more (simulation) time for probing their neighbours. as introduced to speed up the network joining process, allowing all nodes to join the 6TiSCH network successfully and thus to start the LQE strategy as soon as possible.

For the other 6TiSCH protocol parameters, we keep the default values either specified in standards documents [73], [86] or set in the Contiki implementation, as summarised by Table 5.3. The sole exception concerns the ETX value used for the link initialization (ETX_{Init}): we set $ETX_{Init} = 5.0$ as the unknown quality of any

Tab. 5.2: Network topology metrics for the simulated scenarios

	Random-15	Grid-5x5
Nr. nodes	15	25
Nr. very good links	40	40
Nr. good links	23	32
Nr. intermediate links	12	0
Nr. bad links	30	78
Avg. very good or good links	8.4	5.76
Max. very good or good links	13	8
Min. very good or good links	5	3
Avg. hops	1	2
Max. hops	2	4
Avg. path ETX	1.37	2.98
Max. path ETX	2.21	4.60

link before probing [2]. As mentioned in Section 3.3.2, with this choice we aim to prevent possible side-effects caused by a biased link metric initialization as this value is a compromise between the lowest and the greatest possible ETX in the network (i.e. $ETX_{min} = 1$ for $PDR = 1.0$ and the worst case $ETX_{max} = 8$ related to the maximum number of retransmissions set by the MAC layer before discarding the frame). Moreover, this greater value for ETX_{Init} leads to a strategy diametrically opposed to the one adopted in [207], where the authors assume any unknown links as good (by setting $ETX_{Init} = 1$), and, thus, force nodes to keep switching among potentially new good parents. As stated earlier in Section 5.2, parent changes in 6TiSCH networks are energy and time-consuming events that a node should do with care, i.e. at least based on an accurate estimation of the link quality to this candidate new parent node.

We study the effect of the probing periods t_{pp} and the number of shared slots N_{ss} on the performance of the passive and active LQE techniques described earlier. In doing so, we evaluate at the end of the simulation the *absolute estimation error of ETX* (ΔETX). This is defined as the difference between the nominal and the estimated ETX of a link. All presented results are averaged over 20 independent simulations runs, and are shown together with the $95\,\%$ confidence interval.

[2] $ETX_{Init} = 2.0$ is the current default in the Contiki implementation

Tab. 5.3: Simulation settings

Parameter	Value
Simulation time	15 min
Simulation runs	20
TSCH slotframe length	101
TSCH timeslot duration	10 ms
TSCH number of channel	4
TSCH EB period	16 s
TSCH number shared cell	{2, 4, 8}
RPL minimum interval	128 ms
RPL interval doubling	12
RPL redundancy constant	5
LQE probing period	{0, 15, 30, 60} s
LQE initial ETX	5.0
LQE max ETX	8.0

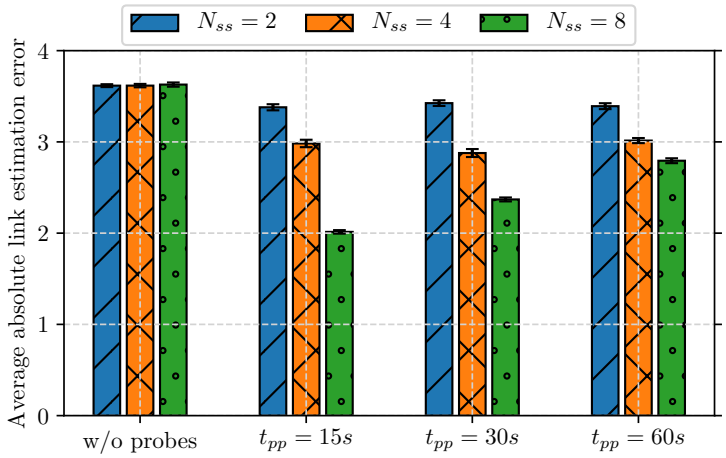
Simulation Results

Figure 5.3 illustrates the ΔETX for the topologies *random-15* and *grid-5x5*. The different colours of the bars stand for the three different considered number of shared slots (i.e. $N_{ss} \in \{2, 4, 8\}$) allocated at bootstrap in the *minimal schedule*. For both topologies, we observe a ΔETX always higher than 3.5 when the default passive strategy (i.e. *w/o probes*) is used, irrespective to the number of minimal cells instantiated. Obviously, enabling the active link probing mechanism reduces the magnitude of the error; this can be seen by comparing the leftmost bars to the rest of the figure. However, even after $15\,\mathrm{min}$, the average ΔETX remains always above 2 in both topologies. As also highlighted in [188], this erroneous LQE can lead to a very unstable topology during bootstrap, influencing the overall reliability and delaying the steady-state of the network; thus, this magnitude of the error is not acceptable. Nevertheless, Figure 5.3a and Figure 5.3b indicate that deploying more timeslots at bootstrap helps in reducing the magnitude of the error on the estimation of the ETX, when RPL uses the active probing strategy (with different probing periods t_{pp}). This fact does not only stem from a reduced 6TiSCH network formation time [3] (reported for the sake of completeness in Table 5.5), but mostly from decreasing the probability that probing messages collide on shared cells. Although probes in RPL are unicast messages, nodes may only transmit them using shared cells at network bootstrap, contending with all the other nodes in the neighbourhood. We observe in both scenarios, how the collisions experienced by the probing messages with $N_{ss} = 2$

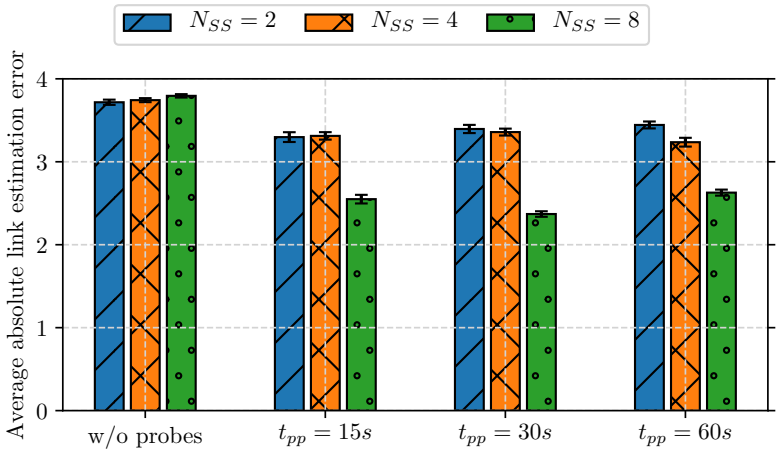
[3] Moving from $N_{ss} = 2$ to $N_{ss} = 8$ enlarges the time windows for probing of ca. $100\,\mathrm{s}$, i.e. nodes may send on average only three additional probes when $t_{pp} = 30\,\mathrm{s}$

or $N_{ss} = 4$ limit even the accuracy of the state-of-the-art LQE mechanism in RPL at network bootstrap. Finally, note that increasing the value N_{ss} brings any advantage for the default passive strategy (c.f. the leftmost bars in the figures). The reason is that, even if nodes discover more links thanks to the better diffusion of control messages at bootstrap (c.f. Section 4.3.3), they actually assess the quality of only a small set of them; the rest retains the initial unknown quality $ETX = ETX_{init}$.

All in all, our results allow to conclude that the ETX values obtained using these shared slots are more a rough indication of the congestion on this broadcast channel than a reliable metric for building the network topology or for scheduling dedicated cells.



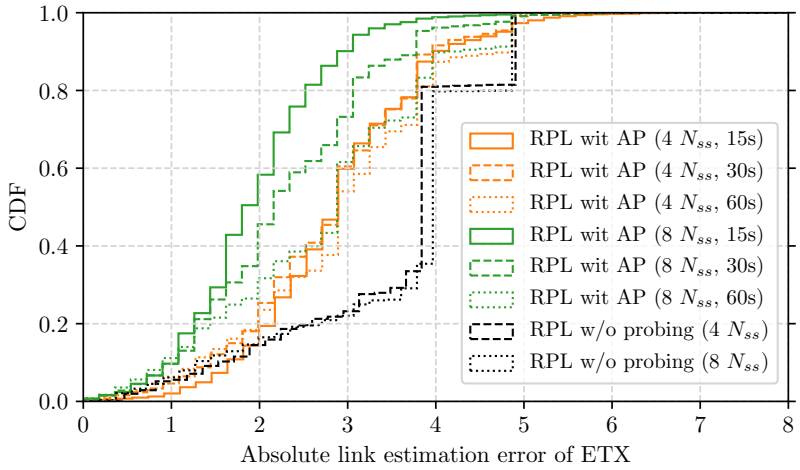
(a) Topology *random-15*



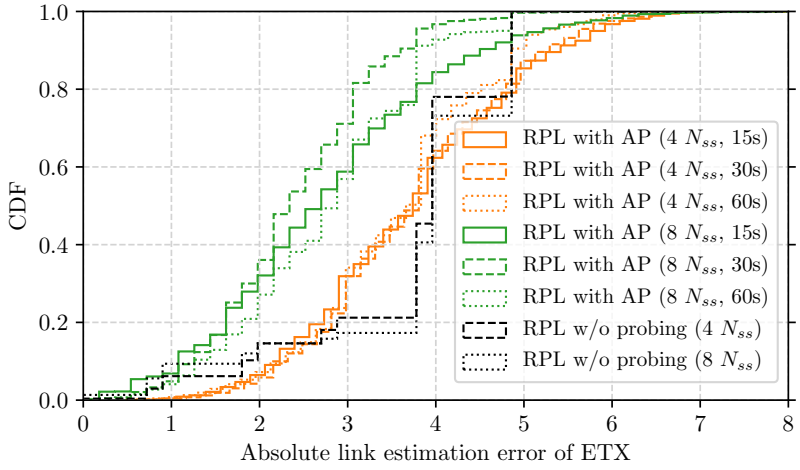
(b) Topology *grid-5x5*

Fig. 5.3: Average absolute estimation error of ETX after 15 min

To get a more detailed insight on different accuracies obtained during the simulation, we show the *Cumulative Distribution Function* (CDF) of the absolute link estimation error in Figure 5.4. Each subfigure shows results for one considered topology. We observe that the default passive strategy, represented with the thinnest black lines, produces most of the time an estimation which is very far from the nominal link quality we set in the two topologies (e.g. $80\,\%$ of the samples are affected by an ΔETX higher than 3). A more accurate LQE is obtained when the active periodic probing strategy (*RPL with AP*) is adopted (orange and green lines) and the number of shared cells increases. In particular, we recognize the best performance with $N_{ss} = 8$, where less than $40\,\%$ of the errors have a magnitude higher than 3 in *grid-5x5* topology and less than $25\,\%$ in the *random-15* topology.



(a) Topology *random-15*



(b) Topology *grid-5x5*

Fig. 5.4: CDF of the absolute estimation error of ETX

At first glance unexpected is the fact that changing the probing period (i.e. the parameter t_{pp}) in the active RPL strategy produces an almost negligible effect on the estimation accuracy. Indeed, in both simulated topologies, the estimation of the ETX remains inadequate, even if nodes transmits the probing messages more frequently. That is, nodes spent more energy for probing for nothing. Nevertheless, this observation supports our claim about the problems caused by the collisions experienced by the probing messages.

The simulation results showed in this section reveal that even the state-of-the-art for accurate link measurements in RPL does not perform well in a time slotted environment. In the best case (i.e. setting $N_{ss} = 8$, $t_{pp} = 15\,\mathrm{s}$ and considering the topology *random-15*), we measure an average error in the estimation of the link quality equal to 2 and only $50\,\%$ of estimations deviate from the nominal ETX less than 2. We argue that this performance does not guarantee a reliable and stable setup of a 6TiSCH topology. This claim motivate our work of improving LQE accuracy in 6TiSCH network by introducing a novel approach for estimating the ETX of links avoiding the use of shared cells.

5.4 DBS-P Allocation Strategy

The previous section has shown how existing LQE strategies result in a non-negligible estimation error during the network formation procedure. We believe that the drawbacks are mainly related to the default timeslot allocation strategy of the 6TiSCH-MC. Indeed, the 6TiSCH-MC reduces the concept of link in 6TiSCH (see Section 5.2) to a simple broadcast channel used via a slotted-Aloha access procedure. Thus, by collecting link statistics using the *minimal slotframe*, nodes obtain more a rough indication of the throughput on this broadcast channel than a reliable metric for building the network topology or for reserving dedicated cells with neighbours. Moreover, the use of an active LQE strategy with unicast probe packets negatively influences the network formation phase. In fact, with more traffic load injected on the slotted Aloha cell offered by the 6TiSCH-MC, we increase the likelihood of a collision for the control packets responsible for TSCH synchronisation and the DODAG joining in RPL. This is even more so due to the resulting retransmissions of unicast probe packets after their failed transmission attempts.

These observations motivate the design of DBS-P, our proposal for LQE in 6TiSCH. DBS-P consists of three main elements:

- an extension of the 6TiSCH-MC by an additional coexisting slotframe for LQE and the definition of its allocation strategy. This solution allows a collision-free transmission of broadcast probe messages even during network setup
- the definition of a small probe broadcast frame for measuring the forward and reverse delivery ratios of the links in the neighbour table and for estimating the ETX on 6TiSCH link following the original definition in [119]
- a simple cross-layer policy between MAC and Network (NET) layer that supports LQE and shortens the neighbour discovery process in 6TiSCH

In the following, our approach will be described extensively: first, we detail on these three elements; then, we show how DBS-P performs compared to the state of the art of LQE.

5.4.1 Design and Implementation

DBS-P relies on Orchestra [185], the cell scheduling module for TSCH implemented in Contiki. In a nutshell, Orchestra allows each node to compute and update its own, local schedules autonomously, relying on existing network stack information such as the MAC addresses of nodes and their neighbours or local RPL topology information. The Orchestra scheduling module allows to define multiple slotframes of different sizes as standardised in [86]. We use this feature for our DBS-P allocation strategy, and we set two types of slotframes:

(1) MC-SF: This is the default slotframe as defined in 6TiSCH-MC for providing minimal connectivity among nodes. It contains only the *mininimal cell* with the coordinates $(ts = 0, co = 0)$. That is, the first timeslot in the slotframe at channel offset zero is available at each node for both reception and transmission, with the TSCH CSMA/CA algorithm regulating concurrent access.

(2) LQE-SF: This is the additional slotframe specified by our proposal for exchanging probing messages. As a node joins the network, DBS-P sets up an LQE-SF on the node and allocates a set of dedicated cells in this slotframe using the identifiers of the node self and of its neighbours. More precisely, each node has a dedicated TX cell for broadcasting its probe and an RX cell for each known neighbours used for receiving their probing messages.

Let us refer the length of the DBS-P slotframe as L_{LQE-SF}, the identifier of a *node x* as id_x and the set of neighbours of a *node x* as $N(x)$. A *node x* determines the coordinates of its dedicated TX cell using its id_x as

$$(ts_x, co_x)_{TX} = (\,\text{Hash}(id_x) \mod L_{LQE-SF}\,,\,0\,)\ . \tag{5.1}$$

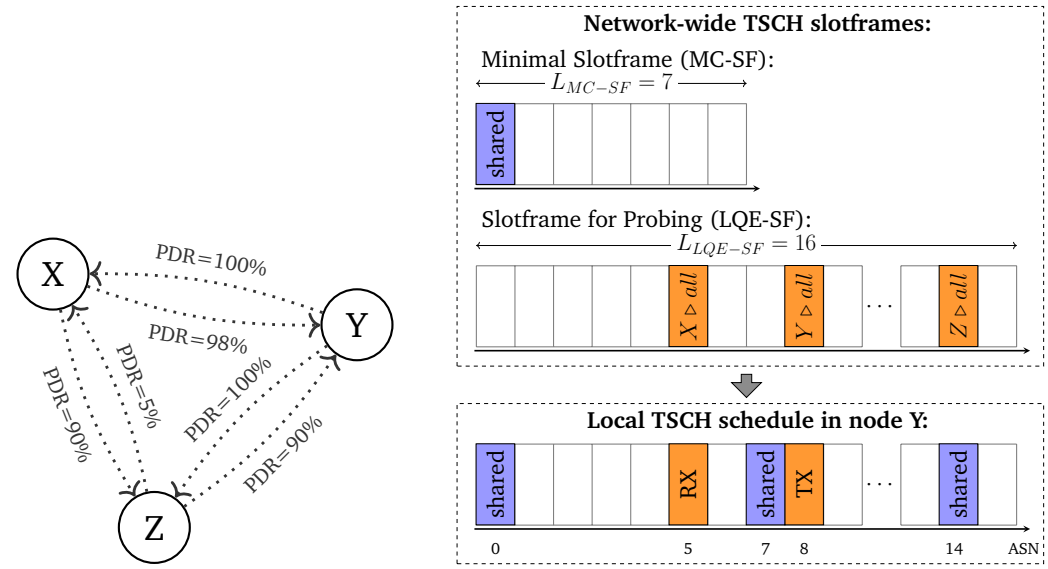
Simultaneously, for each *node n* $\in N(x)$, *node x* computes the coordinate of the RX cell by hashing the neighbour's identifier as

$$(ts_n, co_n)_{RX} = (\,\text{Hash}(id_n) \mod L_{LQE-SF}, 0\,)\ , \tag{5.2}$$

and it knows when it may receive a probe message from *node n*. DBS-P sets with LQE-SF a period $t_{pp} = L_{LQE-SF} \cdot t_s$. Within t_{pp} each node has the opportunity to broadcast a probing message and to update the estimation of the link quality toward all known neighbours.

With the proper choice of the hash function Hash(x), a collision-free transmission of probe messages is made possible by our DBS-P allocation strategy. When (C1) node identifiers are unique, and (C2) L_{LQE-SF} is significantly higher than the number of nodes in the network, then Equation 5.1 can easily produce a unique timeslot index ts_i in every node. On the one hand, the clause C1 applies for node identifiers that are EUI-64-addresses or for 16-bit identifiers. Those are uniquely hard-coded at deployment time, as suggested in IEEE 802.15.4-2015 [86]. On the other hand, the clause C2 dictates that the probing period in DBS-P should not go below a minimum value $t_{pp_{MIN}}$, which is linearly proportional to the network size. However, as explained in details later in 5.4.2, C2 is not a problematic clause for our DBS-P allocation strategy in a typical Industrial Wireless Sensor Network (IWSN). In this work, we use the *modulo function* as Hash(x) together with clauses C1 and C2 for providing a collision-free allocation of the active TX-cells within the LQE-SF.

However, the coexistence of multiple slotframes for delivering various traffic in the network could produce an other problem: the collision between probing messages and other traffic (i.e. Enhanced Beacon (EB) frames, DIO messages or data packets). This could happen when there is an overlap of active cells among these different slotframes, at a specific instant, i.e. at a given Absolute Slot Number (ASN). For example, Figure 5.5b depicts this event at $ASN = 14$, where the *node Y* has simultaneously two cells scheduled, one from LQE-SF and another from MC-SF. To avoid this kind of collision, DBS-P sets the LQE-SF with the lowest priority among other slotframes in the network. Thus, following the policy standardised in [86] (see also 2.4.3), a node will skip the transmission of probe messages, when such an



(a) Network **(b)** TSCH schedules

Fig. 5.5: Illustration of the two slotframes (for minimal traffic and for active probing) in a simple 3-node network with asymmetrical links

overlap of active cells occurs. It is a precise design choice of DBS-P to favour quality over quantity of link measurement.

Note that DBS-P sets a fixed channel offset $co = 0$ in Equations 5.1 and 5.2. However, this choice does not harm our proposal for LQE in the frequency domain, as it may appear at first glance. As already explained in Section 2.4.3, the channel hopping mechanism of TSCH returns different frequencies to be used on successive slotframes, even if the channel offset remains unchanged over time. It is sufficient to select the length of the slotframe for probing L_{LQE-SF} coprime with the number of available channels/frequencies N_c to consider the multi-channel operation of TSCH and measure (at least one time) the quality of each of the N_c channels on a link within a time interval

$$T = N_c \cdot L_{LQE-SF} \cdot t_s = N_c \cdot t_{pp} \ , \tag{5.3}$$

as demonstrated in [109], where t_s is the duration of a timeslot and t_{pp} the probing period.

As above stated, at the instant where a dedicated TX cell in LQE-SF is scheduled, the *node x* broadcasts a probe message, and so it enables all the nodes in $N(x)$ (i.e. neighbours of *node x*) to evaluate the link quality. Considering the simple topology in Figure 5.5, when *node X* broadcasts its probe, *node Y* and *node Z* can measure the *probe reception ratio* for X (i.e. $PrRR_X$), defined as the ratio of the number of probe messages received to the number of probes expected from *X*. The computed $PrRR_X$

represents the reverse delivery ratio d_r of the links $Y \rightarrow X$ and $Z \rightarrow X$, with the notation introduced in Section 2.3.1. The neighbours of *node X* knows the number of probes expected from *X* since the transmission of these messages is periodic. Eventually skipped probing transmissions due to cell overlapping are tracked by a sequence number field transmitted by each node within the probe message.

However, estimating the ETX of a link following the original definition in [119] requires in general not only the reverse delivery ratio d_r but also the forward delivery ratio d_f, as expressed in Equation 5.4 below. To this end, a node includes the number of probe messages previously received from each of its neighbours in the probe's payload.

In the case that a node does not find the counter related to its probing transmission when it receives a probe from a known neighbour, DBS-P recognises the unidirectional nature of this wireless link and makes appropriate actions, e.g. removing the related entry from the neighbour table. In this way, a node prevents the utilisation of this link in the RPL routing topology.

For instance, let us assume that the broadcast probe sent by *node X* contains only a counter related to *node Y* (e.g. $PrCount_Y$) for the network illustrated in Figure 5.5. Then, *node Y* can evaluate the forward delivery ratio d_f for the links $Y \rightarrow X$ and computes the current ETX as

$$ETX = \frac{1}{d_f \cdot d_r} = \frac{PrSent_Y}{PrCount_Y \cdot PrRR_X}\,, \tag{5.4}$$

where $PrSent_Y$ is the number of probe messages sent by *node Y*.

On the contrary, *node Z* decides to discard the link (Z, X), since it recognises this link as not bidirectional.

We use the concept of Information Element (IE) defined in IEEE 802.15.4-2015 [86] to structure the broadcast probe (recall Section 2.4.3 and Figure 2.5). Specifically, we define an LQE sub-IE type following the IETF subtype format described in [100]. Figure 5.6 depicts its format, which we insert as part of the MAC payload in every broadcast probe. We structure the LQE IE as follows. After the canonical subtype ID byte that indicate the presence of this specific sub-IE, it follows a one-byte *LQE control*. There, the first three bits are active flags, and the remaining bits are reserved for any further implementations of the DBS-P.

The active flags define the types of data in the LQE IE as follow: The first flag is the *Asymmetrical Support flag (A-bit)*. When the A-bit is set, then the probe message contains counters related to the neighbours. In this case, our DBS-P takes into

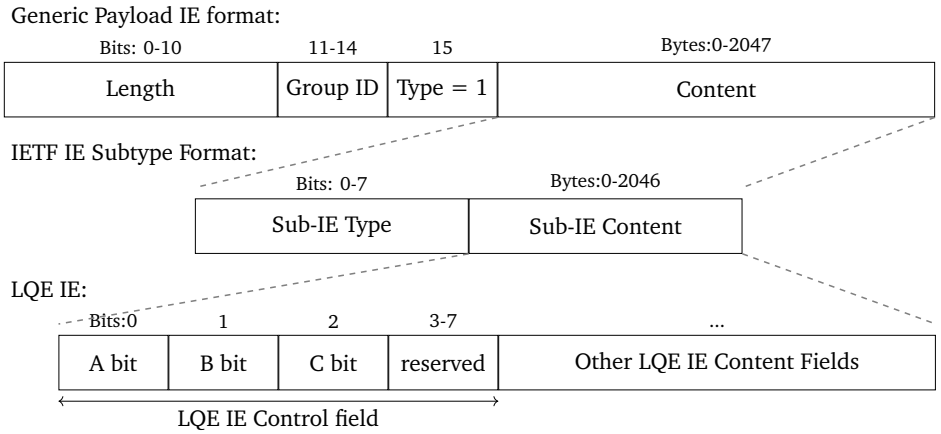


Fig. 5.6: Format of the LQE IE

account the asymmetry in the loss ratios between the two directions of each link, i.e. d_f and d_r. However, evaluating the asymmetry level of links slightly increases both the length of probe messages and the complexity of the calculation of the link's ETX. Thus, for scenarios where the assumption of the symmetry of wireless links is applicable, the A-bit is left unset, and the LQE IE content is merely a sequence number field.

The second flag is the *Bitmap Mode flag (B-bit)*, which defines the mode used to encode/decode an entry in the LQE IE content: Bitmap or Node ID. The *Bitmap mode* (i.e. B-bit set) sorts the counters in the LQE IE content considering the timeslot index in the LQE-SF that each node uses for broadcasting a probe message. We define this mode for reducing the size of the probe message, as no node identifier (EUI-64 or 16 bit short MAC address) needs to be added before each counter for assigning it to a neighbour or link (approach adopted when the B-bit is unset and the *Node ID mode* takes place). Through an allocation bitmap field in the LQE IE content, the Bitmap mode encodes in which cell of the LQE-SF a node has scheduled TX or RX activities related to LQE. Both the length of the Bitmap and the index of the TX cell used by the node for broadcasting its probe are encoded within the first byte in the LQE IE content. The possible lengths of the Bitmap are 2, 4 or 8 bytes. To each set bit in those bytes corresponds a counter-byte in the LQE IE content. Table 5.4 compares the probe message sizes of the Bitmap or Node ID modes and shows the efficiency of the former one when DBS-P operates in small network size (e.g. up to 64 nodes) and dense deployments (e.g. keeping track of at least 16 neighbours).

The last active flag is the *Complete Table flag (C-bit)*, which indicates if the LQE IE contains all the entries in the neighbour table (C-bit set) or only a subset of them (C-bit unset). Including only a subset of links, e.g. the three links with the best d_r

Tab. 5.4: Size of LQE IE content in Bytes.

	tuple *(network size, neighbour table size)*					
Mode	*(32,8)*	*(32,16)*	*(64,8)*	*(64,16)*	*(128,8)*	*(128,16)*
Bitmap	15	23	19	27	27	35
Node ID	25	49	25	49	25	49

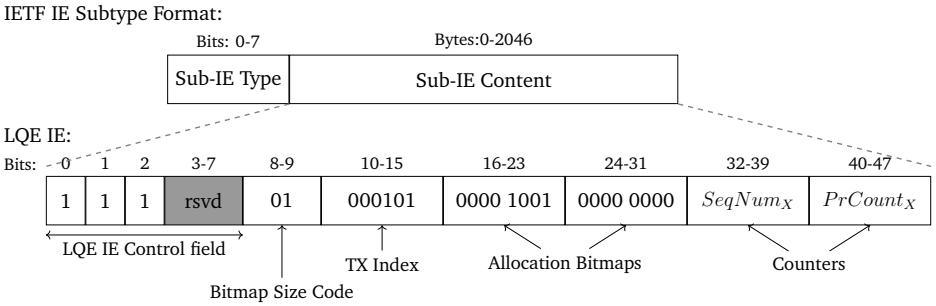


Fig. 5.7: Example of LQE IE content

in the neighbourhood, produces a less accurate neighbour map in each node and may lead to sub-optimal path. Yet, this approach let the LQE mechanism keep track of only the neighbours' set that matters, reducing in this way energy and memory requirements of DBS-P.

In order to better understand the use of this IE in DBS-P, let us introduce an exemplary LQE IE in Figure 5.7. It represents the LQE IE transmitted by *node X* in the simple network of Figure 5.5. In this example, we assume that the hash function in Equation 5.1 returns the following timeslot indexs for the TX cells of the three nodes: $ts_X = 5$, $ts_Y = 8$ and $ts_Z = 14$. We consider here the case that the A-, B- and C-bit are set. Thus, the first two bits encodes the length of the allocation bitmap (in this case, $01 \rightarrow 2$ bytes [4]). The following six bits are the binary representation of the integer 5, i.e. the index in the Bitmap that represents the TX cell used by the *node X* for broadcasting its probes. In the two-bytes long allocation bitmap, also index 8 is set, indicating that also information related to *node Y* are available. Therefore, two counters (each of one byte) are present in the LQE IE: the sequence number of the probe and the number of probe received by *node X* from *node Y*. When *node Y* receives a probe message containing this LQE IE, it recognises its corresponding index in the bitmap (i.e. index 8) as the second bit set. Thus, *node Y* considers the second counter in list as $PrCountY$, and it uses this value for computing d_f and the ETX of the (bidirectional) wireless link to *node X* according Equation 5.4.

[4] Note that these two bits are simply mapped to the three possible lengths of the Bitmap, they are not a binary representation of these lengths. So, $10 \rightarrow 4$ bytes and $11 \rightarrow 8$ bytes

It is known that the sooner a node learns about its neighbours, the better an LQE strategy performs. However, traditional LQE strategies proposed for RPL do not properly make efforts in this direction. Rather, they simply wait the reception of Internet Control Message Protocol for IPv6 (ICMPv6) control packets such as DIO messages (recall Section 2.4.4) or data packets from *node x* for inserting the sender in the neighbour table. This approach would be a limiting factor for LQE in 6TiSCH, especially for the following two reasons. First, both ICMPv6 and data packets are queued with a lower priority than EBs, and so their transmission may be significantly delayed at network bootstrap [191], [233]. Second, the transmission of DIO messages in RPL does not properly have a periodic rate and it is not well-balanced among nodes [216]. Therefore, we introduce a simple cross-layer policy between MAC and NET layer with DBS-P to overcome this issue. Specifically, *node x* adds *node y* in its neighbour table $N(x)$ as soon as it receives a TSCH EB frame from *node y*. Hence, the MAC layer accelerates the neighbour discovery performed at the NET layer by using these control frames, which are transmitted periodically by all nodes already in the network.

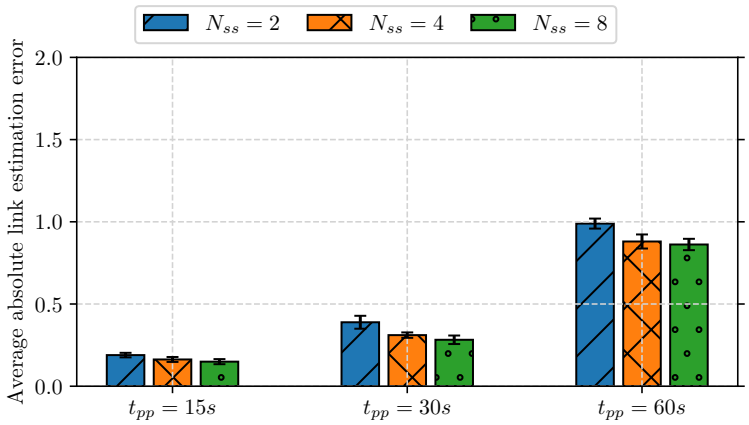
5.4.2 Performance Evaluation

This section presents a detailed evaluation of the DBS-P mechanism introduced in Section 5.4. We begin by evaluating the accuracy of LQE through DBS-P. To this end, we use the simulation setup already presented in Section 5.3.2. Next, we compare DBS-P with the active probing strategy for RPL proposed by Duquennoy et al. [197]. As already stated in Section 5.3.2, this optional strategy represents the state-of-the-art for accurate link measurements in RPL, and it is adopted by the implementation of RPL in Contiki OS (as well as in Contiki-NG). Finally, we investigate the impact of the improved LQE accuracy obtained with DBS-P on the performance of the routing protocol RPL.

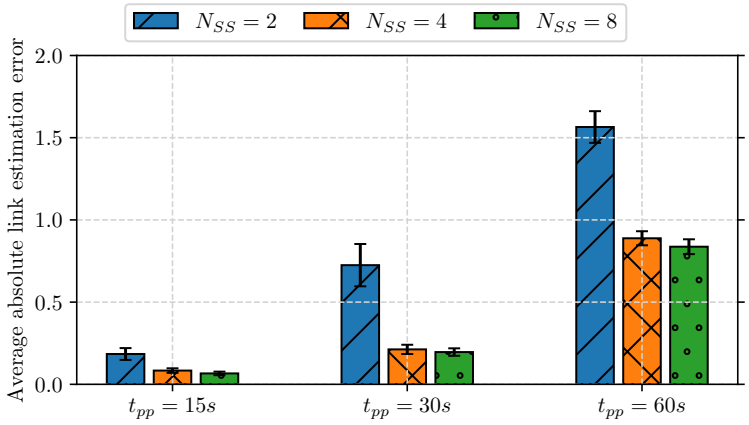
Accuracy of LQE with DBS-P

In Section 5.3.2 we showed the shortcomings of existing LQE strategies on 6TiSCH networks. Here, we use the same simulation setup for evaluating the accuracy of LQE provided by our DBS-P algorithm. Figure 5.8 shows the average difference between the nominal and the estimated ETX (ΔETX) of all known links in the network at the end of the simulation (i.e. after 15 minutes). Figure 5.9 shows the Cumulative Distribution Function (CDF) for all values of ΔETX obtained during

the simulation. Through these figures, we can evaluate the effects of the probing period t_{pp} and of the number of shared cells N_{ss} on the quality of the estimation.



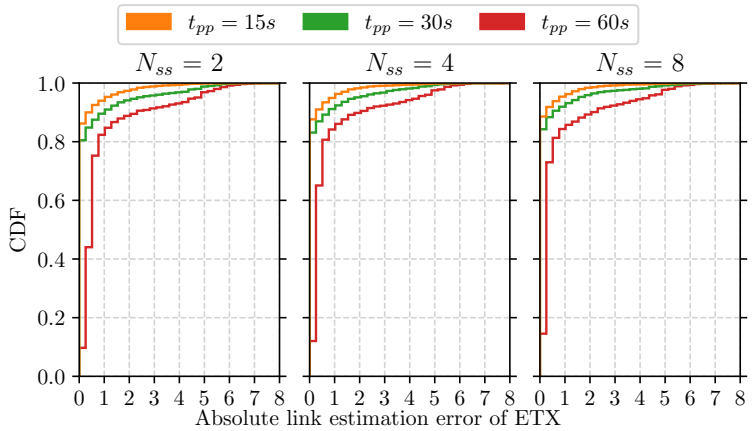
(a) Topology *random-15*



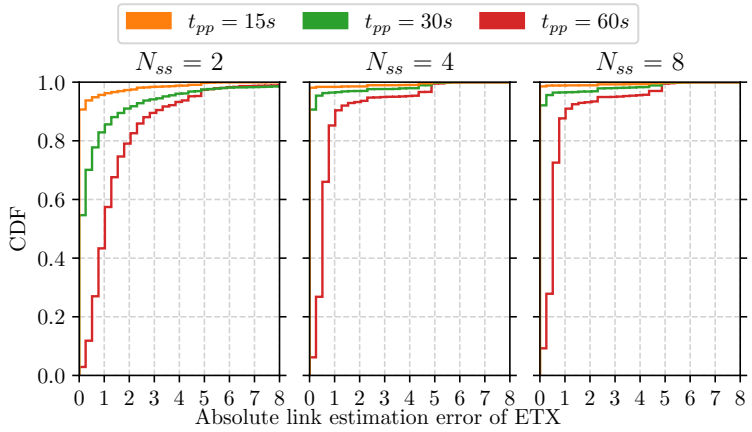
(b) Topology *grid-5x5*

Fig. 5.8: Average absolute estimation error of ETX at end of simulation

First of all, we observe a significant reduction of the magnitude of the error on the estimation of the ETX, if compared to the results presented in Section 5.3 (i.e. the standard methodologies). For all the various combination of t_{pp} and N_{ss}, the average absolut error on the link estimation is always smaller than 2.0. In contrast, the default passive strategy and the simple periodic probing provide values always larger than 3.5 or 2.0, respectively (c.f. Figures 5.3b and 5.3a). The accuracy of the LQE mechanism is particularly good in the simulations with the topology *random-15*, where DBS-P allows for an absolute error smaller than 1.0. This behaviour does not indicate that DBS-P prefers the random topologies over the regular ones. Rather, it is mainly caused by the smaller amount of nodes (i.e. 15 against 25) that allows a faster 6TiSCH network formation and, thus, additional time for link quality estimation than



(a) Topology *random-15*



(b) Topology *grid-5x5*

Fig. 5.9: CDF of the absolute estimation error of ETX

in the grid-5x5 scenario (recall the rule *the sooner a node learns about its neighbours, the better an LQE strategy performs* mentioned in the previous section).

In contrast to the results shown in Section 5.3, the quality of the estimation depends on the probing period t_{pp}. It improves as the t_{pp} reduces since each node broadcasts the probing messages more frequently. Moreover, we recognise the positive side effect obtained by increasing the number of shared cells in the MC-SF, especially in the topology *grid-5x5*.

To investigate this effect more thoroughly, we additionally consider the 6TiSCH network formation time. This is reported in Table 5.5. Note that increasing N_{ss} from 2 to 4 almost halves the network formation time; thus, all the nodes join the network faster and are able to send probing messages for a longer time during the simulation. The Table 5.5 also shows that a further increase of N_{ss} plays a minor role since

Tab. 5.5: 95^{th} Percentile of the average network formation time (s)

		RPL with AP		RPL with DBS-P	
	N_{ss}	Random-15	Grid-5x5	Random-15	Grid-5x5
$t_{pp} = 15\,s$	2	168.24±34.53	562.17±75.90	102.12±15.39	438.91±46.88
	4	79.90±14.92	407.39±56.75	58.67±6.60	251.97±31.75
	8	66.29±11.05	230.09±27.35	47.69±6.09	178.70±17.04
$t_{pp} = 30\,s$	2	151.69±24.47	659.03±87.73	101.29±13.92	434.97±45.45
	4	84.58±11.26	340.33±46.51	74.47±8.60	243.92±34.18
	8	51.03±8.35	213.35±28.80	45.35±5.51	207.23±24.64
$t_{pp} = 60\,s$	2	151.16±23.72	533.37±81.62	104.70±14.75	471.37±47.84
	4	84.20±12.68	289.25±30.52	70.25±10.26	233.90±16.26
	8	57.72±8.98	192.37±26.82	50.47±7.26	192.13±20.23

Tab. 5.6: 95^{th} Percentile of the number of probe messages sent in the network

	N_{ss}	Random-15	Grid-5x5
$t_{pp} = 15\,s$	2	805.6±7.30	1184.7±17.68
	4	868.55±2.98	1321.2±27.50
	8	815.2±2.82	1284.4±20.60
$t_{pp} = 30\,s$	2	416.25±4.07	549.75±17.76
	4	427.95±1.54	647.6 ±15.46
	8	402.65±1.32	622.3 ±11.90
$t_{pp} = 60\,s$	2	202.4±1.41	267.4±5.24
	4	208.2±0.84	317.3±5.36
	8	195.3±0.55	305.9±5.17

the extra time gained does not permit the transmission of many additional probe messages. In addition, setting $N_{ss} = 8$ leads to more likely overlaps between these shared cells in the MC-SF and the scheduled cells in LQE-SF. In this case, the number of LQE measurements sinks a bit, an effect which is visualised by Table 5.6, which contains the number of probes transmitted in the network during the simulation. In fact, a node skips the transmission of the probing packet at these overlap events, since the probe packets have a lower priority than the minimal traffic. The crucial role of t_{pp} in DBS-P is confirmed by Figure 5.9 that shows the cumulative distribution function of the ΔETX of all processed links in the network during the simulation. As can be seen, the more the t_{pp} decreases, the more the CDF lines move to the upper left corner in the figure, i.e. each node obtains a more precise estimation of the links towards its neighbours.

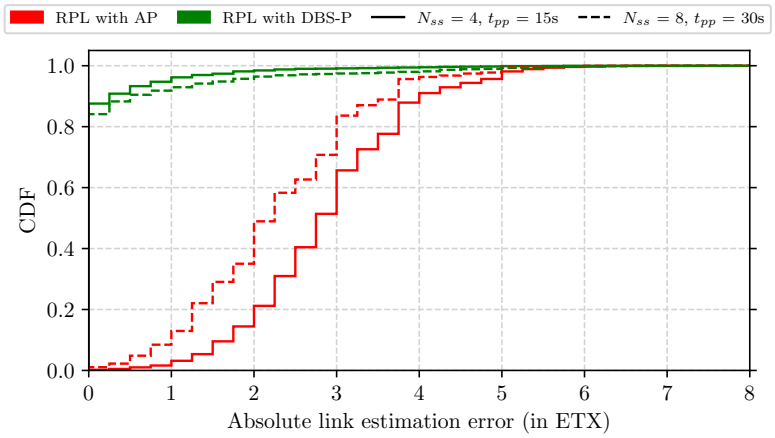
Comparison with the Active Probing Strategy of RPL

After a first proof-of-concept evaluation of the DBS-P mechanism, we compare it with the active probing strategy for estimating the link ETX in RPL proposed by Duquennoy et al. [197] (in the following simply referred as *AP*). This is the active strategy adopted by the RPL implementation shipped with Contiki-NG [11], the newest and widely adopted implementation of Contiki. We have already presented its link estimation algorithm in Section 3.3.2 but also its shortcomings when applied in 6TiSCH networks (c.f. Section 5.3). Figure 5.10 compares the distribution of the absolute link estimation error for both mechanisms in the two simulated topologies. During our simulations, we examined a large parameter design space, but here, we only show the result for the tuple $(N_{ss}, t_{pp}) \in \{(4, 15), (8, 30)\}$, which represents the best parameter set in term of LQE accuracy for the link probing strategies under comparison. We observe that, for both topologies, the curve representing DBS-P lies always clearly above the one representing AP. This means that DBS-P outperforms AP in discovering the neighbourhood during the initial network bootstrap phase in 6TiSCH. For instance, in a regular network with only good or bad links (i.e. Grid-5x5), AP estimates at most only $10\,\%$ of the links with a $\Delta ETX \leq 1$, while our link probing technique at least $90\,\%$.

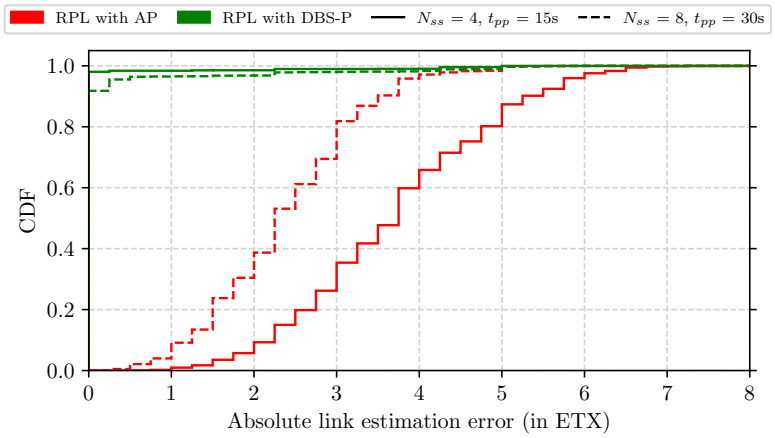
We want to investigate more deeply how good both mechanism help in discovering the neighbourhood of nodes. To this end, we show the number of links in the neighbour table of a node over the time (*nr. of processed link*) and how many of these links are estimated with an absolute error less than 1 (*nr. of accurate links*) in the Figure 5.11. Besides averaging the performance metrics obtained in all nodes, we focus on two particular nodes: *node 7* in the topology *random-15* and *node 13* in the topology *grid-5x5*. For these two nodes, neighbour discovery (especially an accurate one) is challenging as they are the nodes with the highest degree in the two considered scenarios (respectively 14 and 12 [5]).

Although RPL with AP allows the node to discover the presence of almost all the neighbouring nodes, it is not able to estimate the quality of the links towards these neighbour as good as our mechanism. That is expressed by the different distances between the solid and dotted lines in Figure 5.11. As can be seen,those lines converge after an initial delay when RPL adopts the DBS-P mechanism (red lines); in contrast, they stay apart from each other for RPL with AP.This behaviour can be explained considering that the active probing strategy in RPL (i) selects only one target node for probing every t_{pp} and (ii) transmits the unicast probe packet using

[5] here we exclude bad links, i.e. links with a Packet Delivery Rate (PDR) lower than 0.35 (recall Table 3.1)



(a) Topology *random-15*

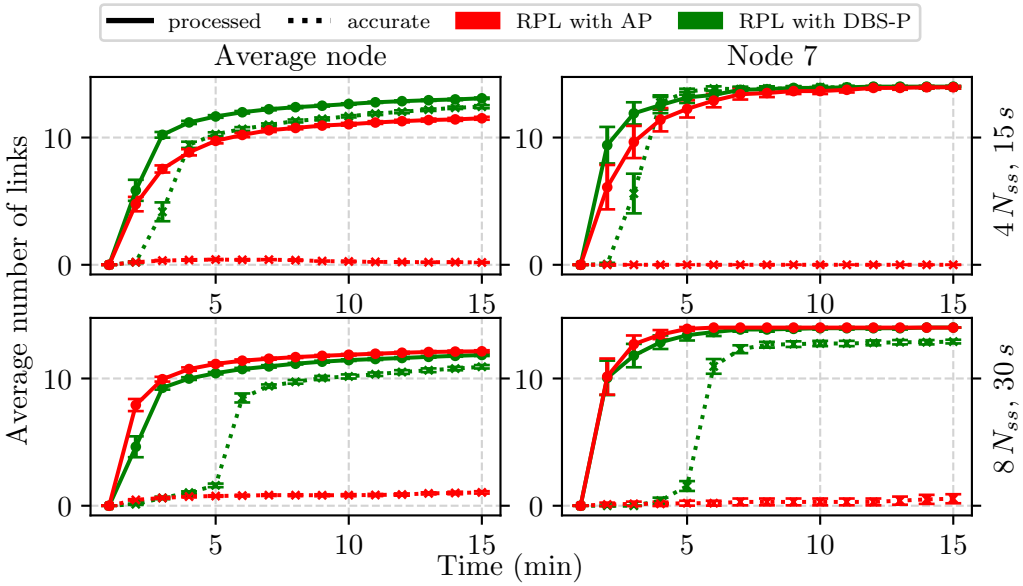


(b) Topology *grid-5x5*

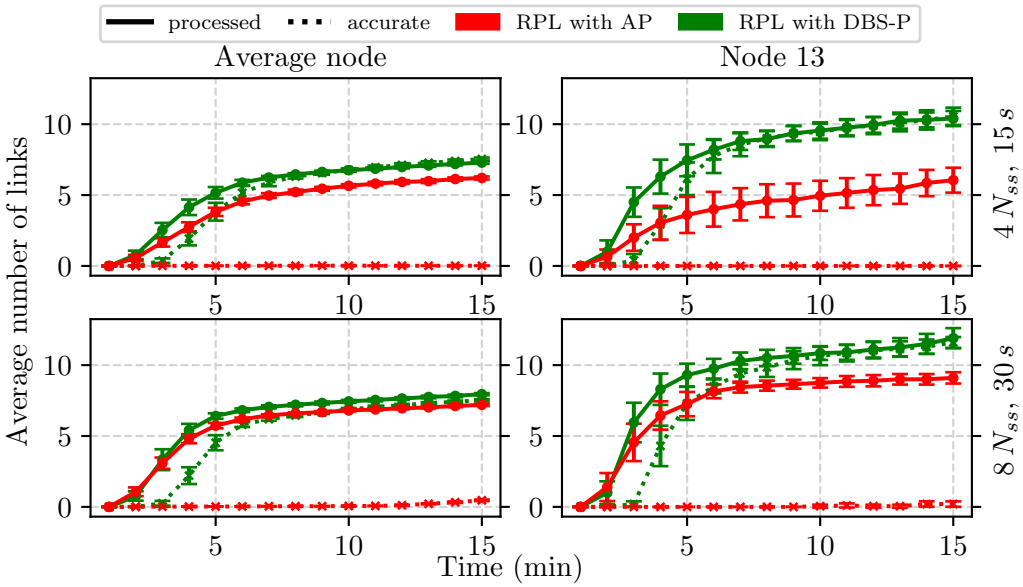
Fig. 5.10: Comparison of RPL with AP and RPL with DBS-P in term of CDF of the absolute estimation error of ETX

shared cells, where collisions may occur. With our proposed probing strategy, in contrast, each node broadcasts a probe messages using a dedicated TSCH cell, which matches a scheduled Rx activity in each neighbouring node. Therefore, each node is able to measure the link quality towards all known neighbours during an interval t_{pp}, without estimation errors caused by collisions. It is this design choice in our DBS-P mechanism that lets nodes accurately estimate each link which is processed (i.e. the lines for processed and accurate links converge in the figures).

Finally, note that our mechanism shows the shortest convergence time with the probing interval $t_{pp} = 15\,\mathrm{s}$ and when the overlaps between LQE-SF and MC-SF are negligible (i.e. $N_{ss} = 4$).



(a) Topology *random-15*



(b) Topology *grid-5x5*

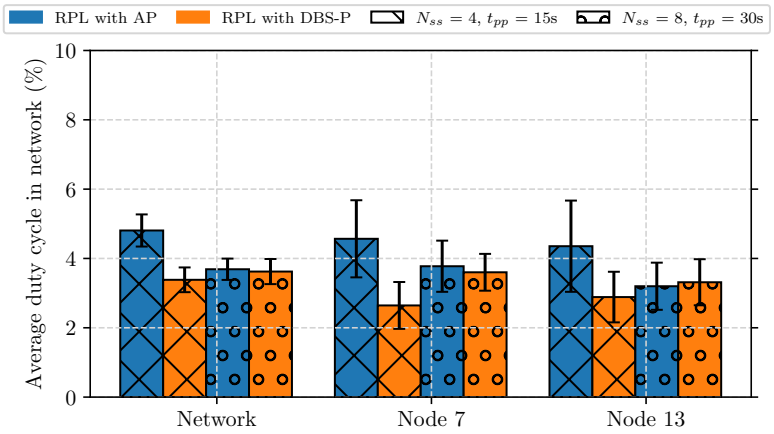
Fig. 5.11: Comparison of RPL with AP and RPL with DBS-P in terms of number of processed and accurate link over time

So far, we have compared the performance of the considered LQE mechanism in term of accuracy. However, a standard criticism about our DBS-P strategy could be that the accurate LQE is obtained by introducing several expensive extra Tx/Rx activities in the network, due to the additional LQE-SF and the transmission of dedicated probe messages. Therefore, we give now an insight on the energy cost required for LQE with DBS-P and RPL-AP mechanisms. To this end, we measure the percentage of time spent by a node in the active state (i.e. transmission, reception or scan activity) during the simulation. This metric evaluates indirectly the energy consumption of the nodes and allows us to compare both mechanisms as in Figure 5.12 (again for both scenarios). Considering the best parameter set for the RPL with AP (i.e. $N_{ss} = 8$ and $t_{pp} = 30\,\mathrm{s}$), RPL with DBS-P has an almost identical energy consumption when compared with RPL-AP (nearly equal in the topology *Random-15* or slightly higher in the topology *Grid-5x5*). However, our strategy consumes far less power than RPL with AP, when using the parameter tuple $(N_{ss} = 4, t_{pp} = 15\,\mathrm{s})$. This behaviour is a bit unexpected considering the greater amount of cells that DBS-P allocates at each node for TX/RX activities within the LQE-SF. The fact is that RPL with AP has a very inefficient behaviour in 6TiSCH networks for this selected parameter set. Recall from Section 5.3.2 that reducing the probing period to 15 seconds does not improve the ETX accuracy as shown in 5.3.2; instead, it increases the likelihood of collision of the control messages for the network formation with the unicast probing packets. This is particularly true when only $N_{ss} = 4$ shared cells are available. Additionally, these collisions produce re-transmissions of unicast probing messages, leading to a vicious cycle that negatively influences both the 6TiSCH network formation time and node's energy consumption.

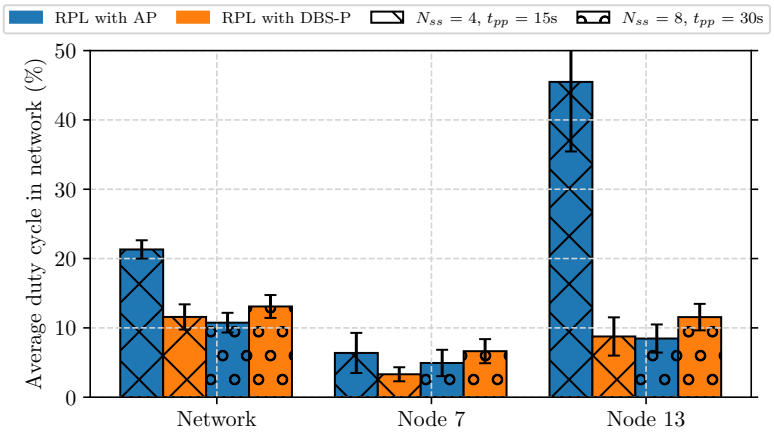
Impact on RPL Performance

As mentioned in Section 3.3.2, an accurate characterisation of link ETX to neighbours is a prerequisite for RPL to select network routes that maximise both network lifetime and throughput. To have an insight on the quality of the route selected by the nodes, we compute the *node stretch* at the end of the simulation (i.e. after $15\,\mathrm{min}$). This metric represents the difference between the cost of the route selected by a node (in terms of overall ETX) and the cost of the ideal shortest path [216].

Figure 5.13 shows the distribution of this metric in both topologies for the two parameter set adopted for the comparison of RPL with AP or with DBS-P. In both scenarios, using RPL with DBS-P limits the stretch of all the nodes to values lower than 0.5, i.e. nodes make a proper route selection. In contrast, using active probing (red lines), RPL computes paths subject to potentially significant stretch. In particular,



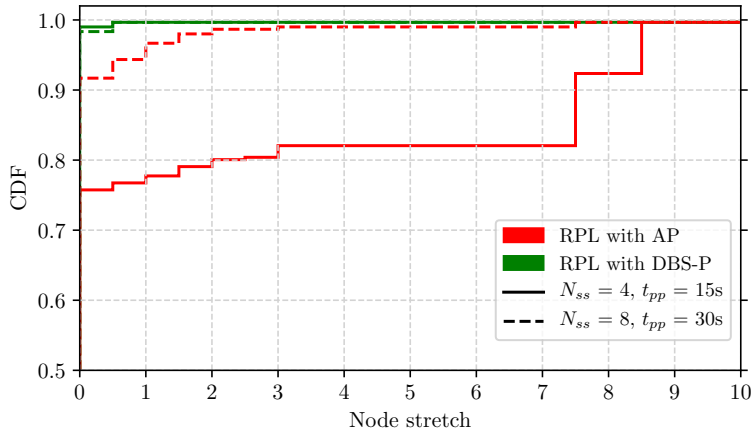
(a) Topology *random-15*



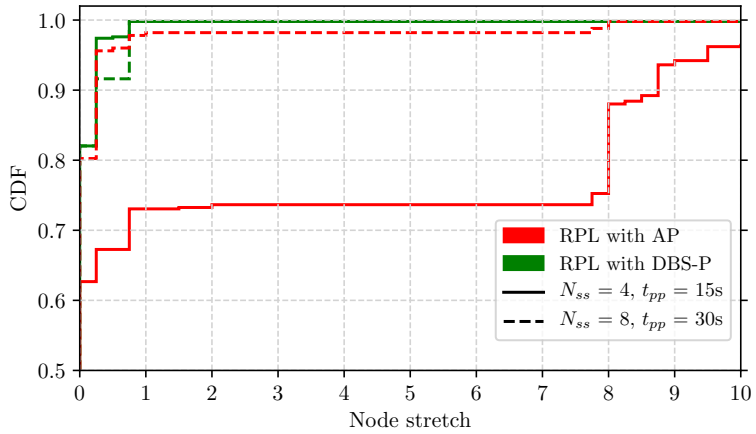
(b) Topology *grid-5x5*

Fig. 5.12: Comparison of RPL with AP and RPL with DBS-P in terms of average duty cycle during the simulation

the relevance of the adopted parameter set is evident. Note that the performance difference in terms of stretch between RPL with AP and RPL with DBS-P appears marginal in the regular topology *grid-5x5* when the parameter $N_{ss} = 8$ and $t_{pp} = 30\,\mathrm{s}$ are considered (i.e. dashed lines). As can be seen, more then $95\,\%$ of node stretches are lower than 1 even when node uses active probing for LQE. This result is a bit unexpected if we remember how less accurate RPL with AP estimates the ETX of links, compared to RPL with DBS-P (c.f. Figure 5.10b). We explain this fact considering that, in this regular topology, a node will choose with high probability the ideal preferred parent, even when it has an inaccurate ETX estimation of the link towards him. This is because the ideal preferred parent often coincides with the emitter of the first DIO message a node receives (considering how the nodes progressively joins the DODAG one by one from the sink located in the upper left



(a) Topology *random-15*



(b) Topology *grid-5x5*

Fig. 5.13: Comparison of RPL with AP and RPL with DBS-P in term of distribution of the node stretch at the end of the simulations

corner in such a regular topology). Since all the neighbours are estimated equally quite bad, a node will then rarely change the preferred parent. However, for opposite reasons, a proper route selection is more challenging in the topology *random-15*. Therefore, here, the comparison between the stretch obtained with active probing (red lines) and with DBS-P (green lines) shows how the latter helps in selecting better routes.

We further analyse the impact of LQE on the performance of a 6TiSCH network by considering the *network functional time* as the time needed by RPL to obtain a topology in which every node has discovered at least a *good link* (recall Table 3.1) to communicate with. In other words, the 6TiSCH network is not only *formed* (see the definition of the performance metric *network formation time* in Section 4.3.2), but it is also *functional*, i.e. bad neighbours that provide poor reliability were filtered out,

Tab. 5.7: 95^{th} Percentile of the network functional time (s)

	N_{ss}	Random-15	Grid-5x5
$t_{pp} = 15\,s$	4	324.40±24.59	470.84±37.54
	8	295.45±19.18	417.25±25.77
$t_{pp} = 30\,s$	4	541.08±37.94	701.25±45.87
	8	499.47±32.21	649.17±34.05

and RPL is converging towards the optimal paths. We consider the *network functional time* as a sort of green light signal in 6TiSCH for starting the process of building and maintaining the TSCH schedule. For instance, the negotiation of dedicated cells among neighbouring nodes through the 6top Protocol (6P) or a Scheduling Function (SF) as proposed in [232]. Both rely on the link quality estimation for computing the required cells for data communication. Here, the dynamic of the RPL protocol in the initial phase or an inaccurate estimation of the ETX between neighbours would otherwise negatively affect the performance of the scheduling algorithm and produce large delays and packet drops in 6TiSCH networks.

As already seen in Figures 5.11b and 5.11a, RPL with AP does not produce an accurate LQE for the neighoburing nodes. For this reason, we could not measure a *network functional time* with merely active probing, even extending the simulation time to 30 minutes. This means that 6TiSCH SFs have to cope with inaccurate ETX values when merely the the 6TiSCH-MC is adopted for network bootstrap.

In contrast, activating DBS-P for LQE in RPL lets the entire network be operational within the simulation time of 15 minutes. We illustrate this fact by showing in Table 5.7 the 95th percentile of the network functional time and in Figure 5.14 the distribution of the node functional time with DBS-P. Again we see that increasing the number of sharedN_{ss} and reducing the probing period t_{pp} have a beneficial effect on the convergence to an accurate estimation of good links. This effect holds for both considered topologies.

A link quality mechanism for 6TiSCH should also avoid excessive RPL preferred parent changes. The reason why this should be avoided is, that when a node switches its preferred parent, it generates more DIO messages due to Trickle timer reset [184]. Besides, a parent change can be considered a disruptive event in 6TiSCH networks. Indeed, it forces nodes to reconfigure their schedule [232] and leads to periods of turbulence in the network [64]. Therefore, we complete our comparison between the active probing strategy and the DBS-P mechanism by showing this facet, i.e. the *number of RPL parent changes* during the simulation in the simulated topology *grid-5x5*. For a fair comparison, we consider only the parameter tuple

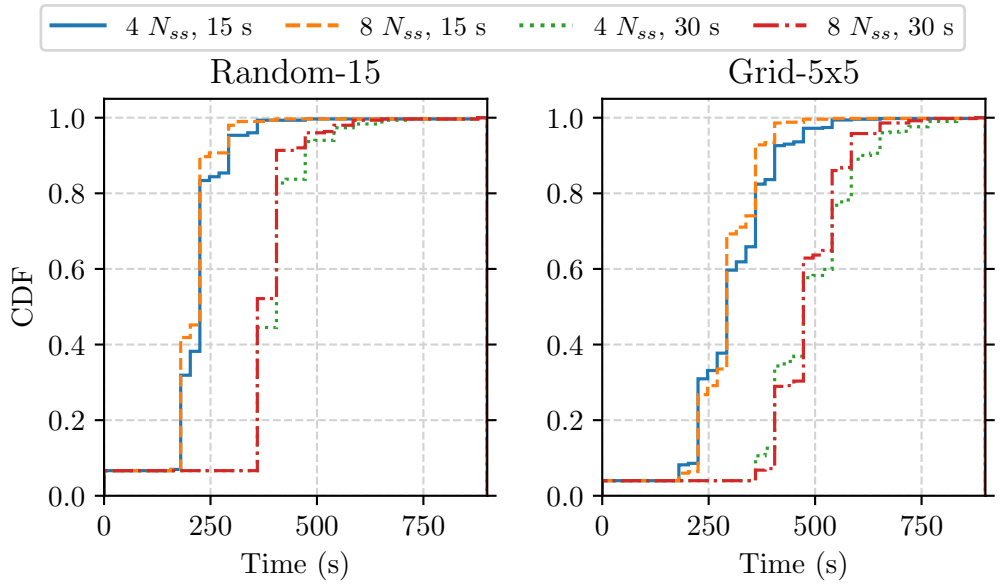


Fig. 5.14: Distribution of node RPL functional time with DBS-P

($N_{ss} = 8, t_{pp} = 30$), which leads for both probing strategies to a comparable *network formation time*. As can be seen in Figure 5.15 (left), the DBS-P mechanism triggers the change of preferred parent more frequently than the active probing strategy in RPL, and, therefore, it introduces more instability in the network. However, we retain that this drawback is not dramatic. First, we observe in Figure 5.15 (right) a huge amount of parent changes before the network is formed. As soon as the nodes have a precise view of the environment (i.e. within 10 minutes), they do not change the preferred parent any more. In sum, the network instability is temporarily confined [6]. Nevertheless, RPL with AP lets node select the preferred parent almost blindly due to the high error rate in the link quality estimation during simulation. The active probing strategy would also trigger several additional parent switch when the estimated ETX of different links would become more accurate (c.f. Figure 5.11).

To confirm the above hypothesis, finally, we show in Figure 5.16 the evolution over time of the RPL rank of the *node 13* for both mechanisms in the simulated topology *grid-5x5*. We select *node 13* as it is the central node in the considered topology. Nevertheless, we observed the same behaviour for the other nodes. Recall from Section 2.4.4 that the rank of a node estimates its distance (here in term of ETX) from the coordinator (i.e. *node 1*). While oscillating more heavily, after some time, our DBS-P mechanism lets *node 13* move closer to the true rank. Both the more

[6] under the assumption that there are no link oscillations as in the simulated scenarios

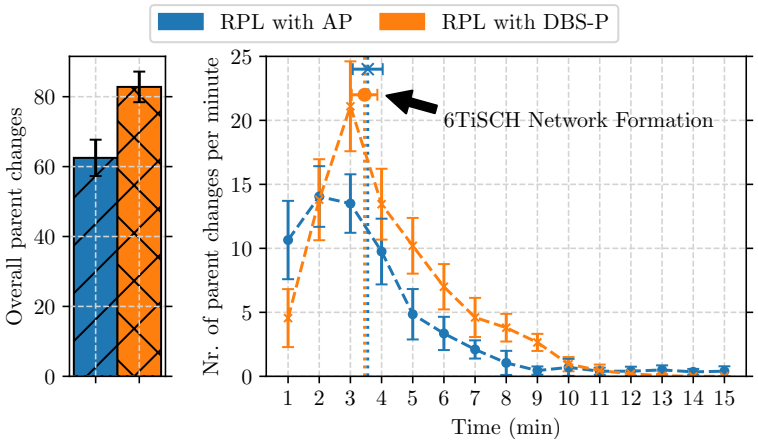


Fig. 5.15: Comparison of RPL with AP and RPL with DBS-P in term of parent changes

frequent parent changes and a more accurate ETX estimation of DBS-P are the reasons of this behaviour.

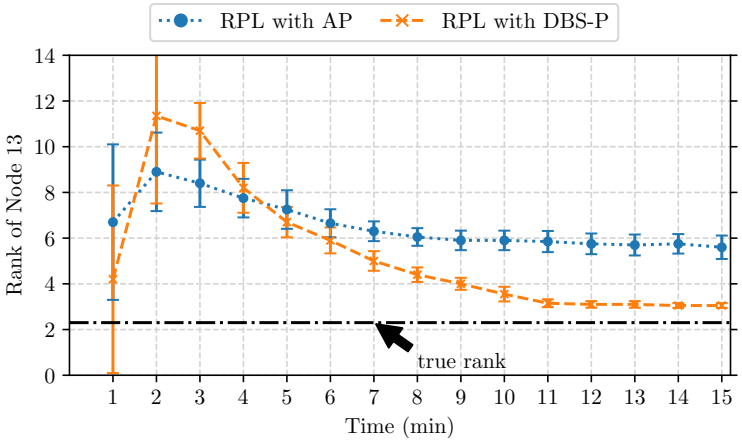


Fig. 5.16: Evolution over time of rank of *node 13* in the topology *grid-5x5*

5.5 Chapter Summary

This chapter has investigated the performance of LQE strategies in 6TiSCH networks. Estimating the link quality is of prime interest in 6TiSCH to build both the best routing path between nodes and network-wide schedules that provide latency and reliability guarantees. A large volume of published studies addressing an accurate and stable LQE for assisting the routing protocol RPL exists. However, these LQE strategies are not designed having TSCH in mind and do not obviously perform in

the same way on 6TiSCH networks. Despite this, LQE has so far received very little attention in the 6TiSCH community.

Firstly, this chapter demonstrated the fallacy of adopting popular LQE strategies on 6TiSCH networks. To this end, we carried out a set of simulations to show that existing passive and active monitoring strategies, which provide accurate link metrics with traditional CSMA/CA-based protocols, do not perform well in 6TiSCH networks, especially during the crucial initial phase. We showed that the default timeslot allocation of the 6TiSCH-MC (i.e. only shared cells) significantly degrade the estimation of the well-established metric ETX.

Secondly, we presented our DBS-P strategy that allocates at network bootstrap an additional slotframe for probing. This solution coexists with 6TiSCH-MC and allows a collision-free transmission of broadcast probe messages even during network setup. DSB-P defines the probing strategy and the content of the small probe messages so that each node has the opportunity to measure the forward and reverse delivery ratios (i.e. compute the ETX metric) of the link toward all known neighbours within a probing interval of less than a minute.

Finally, we showed through performance evaluation that the proposed strategy improves LQE accuracy dramatically at network bootstrap. First of all, we observed a significant reduction of the absolute error on the ETX estimation compared to the results provided by the state-of-the-art for accurate link measurements in RPL. The results showed that DBS-P estimates at least $90\,\%$ of the links with a $\Delta ETX \leq 1$ in different network topologies and in a short time, i.e. within ca. $5\,\mathrm{min}$. The improved accuracy obtained with DBS-P does not have energy overhead as a drawback, as we showed by comparing the average duty cycle of DBS-P with the one of active probing. Moreover, we demonstrated how DBS-P supports nodes in making a proper RPL route selection, i.e. it limits the stretch of nodes to values lower than 0.5 and bounds the RPL oscillations in a restricted time window of only $10\,\mathrm{min}$ at network bootstrap.

6 Conclusion and Outlook

We conclude this thesis highlighting our main scientific contributions and pointing out future research directions related to the work presented here.

6.1 Summary and Contributions

This thesis was dedicated to the study of control primitives such as *network advertising, joining, synchronisation, neighbour discovery* and *topology establishment* that assure a reliable bootstrap and a precise link quality estimation in 6TiSCH networks. We focused our efforts on 6TiSCH, the protocol suite proposed by IETF, as it represents the current leading standardisation effort to enable timed and reliable data communication within IPv6 networks for industrial applications. Above all, the adoption of 6TiSCH promises interoperability between vendors and seamlessly integrates industrial wireless sensor networks into the Internet.

In the 6TiSCH academic community, the design of algorithms for scheduling data transmissions (i.e. 6TiSCH scheduling functions) is at the centre of interest. However, most of the proposed 6TiSCH scheduling functions work with the assumption that the network is already operational or nodes are aware about link and network characteristics. In contrast, this thesis has investigated mechanisms for exchanging the control messages necessary to make a 6TiSCH network operational. Therefore, we mainly dealt with the 6TiSCH network formation process, and we investigated how (i.e. which nodes, what content) and when (i.e. the frequency) the control messages should be transmitted. This is not only important for obtaining a fast joining procedure, but also to enable the scheduling algorithm and to make routing decision possible. While reliability and timeliness in exchanging control messages were our primary goals, we also kept an eye on the conceptual design and we assured that the proposed solutions do not significantly reduce the available bandwidth for data application and the lifetime of power-constrained devices, as attested by the presented performance evaluations.

The scientific contributions of this thesis are threefold:

1. We investigated the mutual interactions between the TSCH scheduling problem and the multi-hop RPL route computation during the network bootstrap phase, highlighting the necessity of a slight modification of the standardised procedure proposed by the IETF. In particular, our investigations have confirmed that the blind adoption of the default configuration of TSCH and RPL protocols lead to a very long (or even unsuccessful) 6TiSCH network formation in different typical topologies. First, this issue derives from the recommended settings for the parameters that govern the generation and transmission rate of EB and DIO messages; indeed, it leads to collisions, buffering or dropping of these control messages. Second, the single shared cell provided by the *minimal slotframe* becomes quickly congested in dense networks, and this fact hinders a proper diffusion of EB and DIO messages. We consequently discussed some recommendations for an appropriate setting of TSCH and RPL parameters.

2. We proposed Multiple Advertisement Cell Allocation (MACA), a scheduling mechanism for coordinating the transmission of control messages among neighbouring nodes in a dynamic and distributed way. With MACA, we have enhanced the 6TiSCH Minimal Configuration (6TiSCH-MC) by allocating at least $N_{adv} \geq 2$ advertising cells, regularly spaced within the slotframe at a specific channel offset. A distinguishing feature of our proposal is the feedback-based reservation of the active cells. This allows each node to autonomously set N_{adv} and select one of them for network advertising by considering the local contention in the neighbourhood. In doing that, no additional control messages are exchanged, but only a few bytes are appended in the anyway transmitted EB and DIO packets. Our proposed solution has achieved a significant reduction of control messages collisions and a more rapid diffusion of DIO messages without complex management overhead or additional energy cost. This, in turn, improves the 6TiSCH network formation in terms of time required by nodes to join the network and energy consumed for this operation in all the considered scenarios. Compared with other possible straightforward extensions of the 6TiSCH-MC, our proposal has emerged as the most appropriate solution in challenging long chain network topologies

3. We addressed the link quality estimation problem in 6TiSCH, demonstrating the inaccuracy of state-of-the-art mechanisms and proposing Dedicated Broadcast Slotframe for Probing (DBS-P), a strategy by which a 6TiSCH node promptly discovers all its neighbours, and accurately estimates the quality of the link (in term of ETX) to that neighbours. To this end, DBS-P introduces an additional TSCH slotframe and meticulously defines the content of probing messages. With this proposal, each node can actively broadcast probing

message without resulting collisions and to measure the asymmetry level of all the links in its neighbourhood in less than a minute. The proposed approach has lead to a substantial reduction of the absolute error on the ETX estimation, without penalty in terms of energy consumption. The improved accuracy obtained with our strategy has, in turn, generated positive implications on the routing process, i.e. allowing it to select the ideal shortest paths and to avoid unnecessary routing oscillations after a stabilisation time of $10\,\mathrm{min}$.

All in all, we see the contributions of this thesis as an enhancement of the standardised 6TiSCH protocol stack for overcoming certain limitations in its control primitives that, in our opinion, may delay or hinder its wide adoption in the industry. Ultimately, our work is essential for quickly obtaining a 6TiSCH network able to deliver data messages in a dependable and energy-efficient way.

6.2 Future Research Directions

Despite its technical depth, the research presented in this thesis can be extended in several interesting directions. Let us now discuss some of them.

Our MACA solution fulfils the goal of improving the 6TiSCH network formation process. We have implicitly considered only the communication paradigm for data collection (i.e. M2P traffic), and we assured that each node could quickly choose a preferred parent for sending the sensed data towards the sink. In contrast, the creation of RPL downward paths for Point-to-multipoint (P2MP) and Point-to-point (P2P) traffics is out of scope from 6TiSCH-MC or our MACA solution as it takes place successively, when joined nodes transmit unicast Destination Advertisement Object (DAO) messages to the root. In general, finding reliable downward routes for RPL is challenging due to link asymmetry, neighbour unreachability detection [234], nodes' memory constraints [235] and poor LQE [197]. Nevertheless, combining together MACA and DBS-P mechanisms for optimising the 6TiSCH network setup for downward traffic seems promising; indeed, the MACA supports the neighbour discovery procedure, and the DBS-P provides accurate ETX estimation taking into account link asymmetry. Therefore, investigating the benefit of MACA and DBS-P on the performance of RPL w.r.t. downward traffic is an intended future work.

Concerning the DBS-P mechanism, we have focussed on an accurate LQE during the network bootstrap phase. As our performance evaluation shows, this phase is in the order of a few minutes, and, therefore, the assumptions that link qualities are static and any data transmission takes place only on dedicated cells are realistic.

However, an interesting future work consists of assessing how our DBS-P mechanism performs in monitoring LQE in 6TiSCH networks during their operational phase, i.e. considering the link dynamics and a cell allocation algorithms that runs in parallel. In this case, it is wise to (i) limit LQE to a subset of good candidate parents (i.e. reducing the number of active RX cells in the slotframe for probing) and (ii) compute ETX aggregating statistics from the different cells (i.e. the ones scheduled for probing and the others for unicast data transmissions).

Another possible direction of future work has to do with the support for mobile nodes, as wearable devices or robots that are deployed in industrial environment together with stationary nodes to form a 6TiSCH network. Their mobility can be either on a micro (i.e. movements inside a 6TiSCH network) or macro (i.e. movements between different adjoining 6TiSCH networks) scale. Obviously, node mobility complicates the network formation process and the link quality estimation in 6TiSCH. On the one hand, the mix of frequent joins and leaves of mobile nodes and slow TSCH synchronisation time may deplete their energy (recall that their radio is always on while joining). On the other hand, the dynamic in each node's neighbourhood complicates selecting the target in LQE active probing strategies. However, the mechanisms proposed in this thesis may partially alleviate these issues. Thanks to the MACA mechanism, a fast joining of mobile nodes is still possible since all control messages are scheduled on a specific frequency on several timeslots during a given slotframe. The DBS-P strategy for LQE adopts broadcast probe messages and the node identifiers for scheduling dedicated TX cells, i.e. without explicit coordination. Nevertheless, introducing distinct handling of mobile and fixed nodes is a useful addition for both MACA and DBS-P mechanisms. This concerns, for instance, the reservation of advertisements cells in MACA or evaluating the asymmetry level of links. Both features are problematic but also unnecessary for mobile nodes.

We believe that the just mentioned research directions can significantly enhance the solutions proposed in this thesis and support the future wide adoption of 6TiSCH in the industry.

List of Abbreviations

4B Four-Bit.

6LoWPAN IPv6 over Low power Wireless Personal Area Network.

6P 6top Protocol.

6TiSCH IPv6 over the TSCH mode of IEEE 802.15.4.

6TiSCH-MC 6TiSCH Minimal Configuration.

6TiSCH-MC 6TiSCH Minimal Configuration.

ADC Analog-to-Digital Converter.

AI Artificial Intelligence.

ASN Absolute Slot Number.

BE Beacon Enabled.

BLE Bluetooth Low Energy.

CCA Clear Channel Assessment.

CDF Cumulative Distribution Function.

CoAP Constrained Application Protocol.

CQE Channel Quality Estimation.

CSMA/CA Carrier-Sense Multiple Access with Collision Avoidance.

DAO Destination Advertisement Object.

DAO-ACK DAO Acknowledgment.

DBS-P Dedicated Broadcast Slotframe for Probing.

DC Direct Current.

DIO DODAG Information Object.

DIS DODAG Information Solicitation.

DMA Direct Memory Access.

DODAG Destination Oriented Directed Acyclic Graph.

DSME Deterministic and Synchronous Multi-channel Extension.

DTMC Discrete Time Markov Chain.

EB Enhanced Beacon.

EEPROM Electrically Erasable Programmable Read-Only Memory.

ETX Expected Transmission Count.

EWMA Exponentially Weighted Moving Average.

GPIO General Purpose Input Output.

I2C Inter-Integrated Circuit.

ICMPv6 Internet Control Message Protocol for IPv6.

IE Information Element.

IEC International Electrotechnical Commission.

IEEE-SA Institute of Electrical and Electronics Engineering.

IETF Internet Engineering Task Force.

IIoT Industrial Internet of Thing.

IoT Internet of Thing.

IPv6 Internet Protocol version 6.

ISA International Society of Automation.

ISM Industrial Scientific and Medical.

IWSN Industrial Wireless Sensor Network.

KA Keep-Alive.

LoRaWAN Long Range Wide Area Network.

LPWAN Low Power Wide Area Network.

LQE Link Quality Estimation.

LQI Link Quality Indicator.

MAB Multi-Armed Bandit.

MAC Medium Access Control.

MACA Multiple Advertisement Cell Allocation.

MANET Mobile Ad Hoc Network.

MCU Microcontroller.

MEMS Microelectromechanical System.

mMTC Massive Machine-Type Communications.

MP2P Multipoint-to-point.

MRHOF Minimum Rank with Hysteresis Objective Function.

MRM Multipath Ray-tracer Medium.

MSF Minimal Scheduling Function.

NB-IoT Narrow-Band IoT.

NBE Non Beacon Enabled.

NET Network.

OF Objective Function.

OF0 Objective Function 0.

OS Operative System.

OSCORE Object Security for Constrained RESTful Environments.

OSI Open System Interconnection.

P2MP Point-to-multipoint.

P2P Point-to-point.

PAN Personal Area Network.

PDP Packet Delivery Probability.

PDR Packet Delivery Rate.

PHY Physical.

PRR Packet Reception Rate.

RAM Random Access Memory.

ROM Read-Only Memory.

RPL Routing Protocol for Low Power and Lossy Networks.

RSSI Received Signal Strength Indicator.

SDO Standard Development Organisation.

SF Scheduling Function.

SNR Signal-to-Noise Ratio.

SoC System on a Chip.

SPI Serial Peripheral Interface.

TDMA Time Division Multiple Access.

TI Texas Instruments.

TSCH Time Slotted Channel Hopping.

UART Universal Asynchronous Receiver-Transmitter.

UDGM Unit Disk Graph Medium.

UDP User Datagram Protocol.

USB Universal Serial Bus.

WG Working Group.

WLAN Wireless Local Access Network.

WSN Wireless Sensor Network.

List of Figures

List of Tables

Bibliography

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] L. D. Xu, W. He, and S. Li, "Internet of Things in Industries: A Survey," *IEEE Trans. Ind. Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[4] T. Watteyne, "The internet of (important) things," PhD thesis, Sorbonne Universite, 2019.

[6] P. Park, S. Coleri Ergen, C. Fischione, C. Lu, and K. H. Johansson, "Wireless Network Design for Control Systems: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 20, no. 2, pp. 978–1013, 2018.

[7] K. Pister, P. Thubert, S. Dwars, and T. Phinney, "Industrial routing requirements in low-power and lossy networks," RFC Editor, RFC 5673, Oct. 2009, `http://www.rfc-editor.org/rfc/rfc5673.txt`.

[9] A. Frotzscher, U. Wetzker, M. Bauer, *et al.*, "Requirements and current solutions of wireless communication in industrial automation," in *2014 IEEE Int. Conf. Commun. Work. ICC 2014*, IEEE Computer Society, 2014, pp. 67–72.

[13] ITU-T Recommendations, "X.200 (1994)—iso/iec 7498-1: 1994," *Information technology–Open systems interconnection–Basic reference model: The basic model*, 1994.

[15] C. Bormann, M. Ersue, and A. Keranen, "Terminology for constrained-node networks," RFC Editor, RFC 7228, May 2014, `http://www.rfc-editor.org/rfc/rfc7228.txt`.

[16] O. Hahm, E. Baccelli, H. Petersen, and N. Tsiftes, "Operating Systems for Low-End Devices in the Internet of Things: A Survey," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 720–734, Oct. 2016.

[17] M. O. Ojo, S. Giordano, G. Procissi, and I. N. Seitanidis, "A Review of Low-End, Middle-End, and High-End Iot Devices," *IEEE Access*, vol. 6, pp. 70 528–70 554, 2018.

[19] H.-S. Kim, M. P. Andersen, K. Chen, *et al.*, "System Architecture Directions for Post-SoC/32-bit Networked Sensors," in *Proc. 16th ACM Conf. Embed. Networked Sens. Syst.*, New York, NY, USA: ACM, Nov. 2018, pp. 264–277.

[20] K. S. Adu-Manu, N. Adam, C. Tapparello, H. Ayatollahi, and W. Heinzelman, "Energy-Harvesting Wireless Sensor Networks (EH-WSNs)," *ACM Trans. Sens. Networks*, vol. 14, no. 2, pp. 1–50, Jul. 2018.

[21] E. White, *Making Embedded Systems*, 2nd. USA: O'Reilly Media, Inc., 2011.

[27] X. Vilajosana, P. Tuset, T. Watteyne, and K. Pister, "OpenMote: Open-Source Prototyping Platform for the Industrial IoT," in *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 155, Springer Verlag, 2015, pp. 211–222.

[28] B. Martinez, X. Vilajosana, I. H. Kim, *et al.*, "I3Mote: An Open Development Platform for the Intelligent Industrial Internet.," *Sensors (Basel).*, vol. 17, no. 5, pp. 1–22, Apr. 2017.

[29] M. P. Andersen, H.-S. Kim, and D. E. Culler, "Hamilton," in *Proc. 4th ACM Int. Conf. Syst. Energy-Efficient Built Environ.*, vol. 2017-Janua, New York, NY, USA: ACM, Nov. 2017, pp. 1–2.

[30] P. Dutta and A. Dunkels, "Operating systems and network protocols for wireless sensor networks," *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 370, no. 1958, pp. 68–84, Jan. 2012.

[31] P. Gaur and M. P. Tahiliani, "Operating systems for IoT devices: A critical survey," in *Proc. - 2015 IEEE Reg. 10 Symp. TENSYMP 2015*, Institute of Electrical and Electronics Engineers Inc., Jul. 2015, pp. 33–36.

[37] A. Dunkels, "Rime — a lightweight layered communication stack for sensor networks," in *Proceedings of the European Conference on Wireless Sensor Networks (EWSN), Poster/Demo session*, Delft, The Netherlands, Jan. 2007.

[39] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd. USA: Prentice Hall PTR, 2001.

[40] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin, "Statistical model of lossy links in wireless sensor networks," in *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, IEEE, 2005, pp. 81–88.

[41] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic, "Models and solutions for radio irregularity in wireless sensor networks," *ACM Trans. Sens. Networks*, vol. 2, no. 2, pp. 221–262, May 2006.

[42] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An empirical study of low-power wireless," *ACM Trans. Sens. Networks*, vol. 6, no. 2, pp. 1–49, Feb. 2010.

[43] C. A. Boano, M. A. Zuniga, T. Voigt, A. Willig, and K. Romer, "The Triangle Metric: Fast Link Quality Estimation for Mobile Wireless Sensor Networks," in *2010 Proc. 19th Int. Conf. Comput. Commun. Networks*, IEEE, Aug. 2010, pp. 1–7.

[44] N. Baccour, A. Koubâa, L. Mottola, *et al.*, "Radio link quality estimation in wireless sensor networks," *ACM Trans. Sens. Networks*, vol. 8, no. 4, pp. 1–33, Sep. 2012.

[45] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An empirical study of low-power wireless," *ACM Trans. Sens. Networks*, vol. 6, no. 2, pp. 1–49, Feb. 2010.

[46] P. H. Gomes, Y. Chen, T. Watteyne, and B. Krishnamachari, "Insights into Frequency Diversity from Measurements on an Indoor Low Power Wireless Network Testbed," in *2016 IEEE Globecom Work. (GC Wkshps)*, IEEE, Dec. 2016, pp. 1–6.

[47] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *2004 First Annu. IEEE Commun. Soc. Conf. Sens. Ad Hoc Commun. Networks, 2004. IEEE SECON 2004.*, IEEE, 2004, pp. 517–526.

[48] K. Woyach, D. Puccinelli, and M. Haenggi, "Sensorless Sensing in Wireless Networks: Implementation and Measurements," in *2006 4th Int. Symp. Model. Optim. Mobile, Ad Hoc Wirel. Networks*, IEEE, 2006, pp. 1–8.

[49] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, pp. 212–225, Oct. 1994.

[50] K. Kredo and P. Mohapatra, "Medium access control in wireless sensor networks," *Comput. Networks*, vol. 51, no. 4, pp. 961–994, Mar. 2007.

[51] A. Bachir, M. Dohler, T. Watteyne, I. Member, and I. S. Member, "MAC Essentials for Wireless Sensor Networks MAC Essentials for Wireless Sensor Networks," *Commun. Surv. Tutorials*, vol. 12, no. 2, pp. 222–248, 2010.

[52] P. Suriyachai, U. Roedig, and A. Scott, "A Survey of MAC Protocols for Mission-Critical Applications in Wireless Sensor Networks," *IEEE Commun. Surv. Tutorials*, vol. 14, no. 2, pp. 240–264, 2012.

[53] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 15, no. 1, pp. 101–120, 2013.

[54] A. Kumar, M. Zhao, K.-J. Wong, Y. L. Guan, and P. H. J. Chong, "A Comprehensive Study of IoT and WSN MAC Protocols: Research Issues, Challenges and Opportunities," *IEEE Access*, vol. 6, pp. 76 228–76 262, 2018.

[55] B. A. Nahas, S. Duquennoy, and O. Landsiedel, "Poster abstract: Network bootstrapping and leader election utilizing the capture effect in low-power wireless networks," in *SenSys 2017 - Proc. 15th ACM Conf. Embed. Networked Sens. Syst.*, vol. 2017-Janua, New York, NY, USA: Association for Computing Machinery, Inc, Nov. 2017, pp. 1–2.

[56] R. C. Carrano, D. Passos, L. C. S. Magalhaes, and C. V. N. Albuquerque, "Survey and Taxonomy of Duty Cycling Mechanisms in Wireless Sensor Networks," *IEEE Commun. Surv. Tutorials*, vol. 16, no. 1, pp. 181–194, 2014.

[57] C. Cano, B. Bellalta, A. Sfairopoulou, and M. Oliver, "Low energy operation in wsns: A survey of preamble sampling mac protocols," *Computer Networks*, vol. 55, no. 15, pp. 3351–3363, 2011.

[58] R. Su, T. Watteyne, and K. S. J. Pister, "Comparison between Preamble Sampling and Wake-Up Receivers in Wireless Sensor Networks," in *2010 IEEE Glob. Telecommun. Conf. GLOBECOM 2010*, IEEE, Dec. 2010, pp. 1–5.

[59] K. Langendoen, "Medium access control in wireless sensor networks," *Medium access control in wireless networks*, vol. 2, pp. 535–560, 2008.

[60] T. Watteyne, B. Kerkez, K. Pister, and S. Glaser, "Crystal-Free Network Synchronization," *Trans. Emerg. Telecommun. Technol. Trans. Emerg. Tel. Tech*, vol. 00, pp. 1–13, 2016.

[61] T. Chang, T. Watteyne, K. Pister, and Q. Wang, "Adaptive synchronization in multi-hop TSCH networks," *Comput. Networks*, vol. 76, pp. 165–176, Jan. 2015.

[62] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "EM-MAC," in *Proc. Twelfth ACM Int. Symp. Mob. Ad Hoc Netw. Comput. - MobiHoc '11*, New York, New York, USA: ACM Press, 2011, p. 1.

[63] R. Soua and P. Minet, "A survey on multichannel assignment protocols in wireless sensor networks," in *IFIP Wirel. Days*, vol. 1, 2011.

[64] R. Teles Hermeto, A. Gallais, and F. Theoleyre, "On the (over)-Reactions and the Stability of a 6TiSCH Network in an Indoor Environment," in *Proc. 21st ACM Int. Conf. Model. Anal. Simul. Wirel. Mob. Syst.*, New York, NY, USA: ACM, Oct. 2018, pp. 83–90.

[65] M. Baddeley, R. Nejabati, G. Oikonomou, M. Sooriyabandara, and D. Simeonidou, "Evolving SDN for Low-Power IoT Networks," in *2018 4th IEEE Conf. Netw. Softwarization Work.*, IEEE, Jun. 2018, pp. 71–79.

[66] J. Eriksson, N. Finne, N. Tsiftes, S. Duquennoy, and T. Voigt, "Scaling rpl to dense and large networks with constrained memory," in *Proceedings of the 2018 International Conference on Embedded Wireless Systems and Networks*, ser. EWSN '18, Madrid, Spain: Junction Publishing, 2018, pp. 126–134.

[67] P. Karkazis, H. C. Leligou, L. Sarakis, *et al.*, "Design of primary and composite routing metrics for RPL-compliant Wireless Sensor Networks," *2012 Int. Conf. Telecommun. Multimedia, TEMU 2012*, pp. 13–18, 2012.

[68] S. Capone, R. Brama, N. Accettura, D. Striccoli, and G. Boggia, "An Energy Efficient and Reliable Composite Metric for RPL Organized Networks," *Proc. - 2014 Int. Conf. Embed. Ubiquitous Comput. EUC 2014*, pp. 178–184, 2014.

[69] T. Watteyne, A. Mehta, and K. Pister, "Reliability through frequency diversity," in *Proc. 6th ACM Symp. Perform. Eval. Wirel. ad hoc, sensor, ubiquitous networks - PE-WASUN '09*, New York, New York, USA: ACM Press, 2009, p. 116.

[70] J. Faruque, K. Psounis, and A. Helmy, "Analysis of gradient-based routing protocols in sensor networks," in *Lect. Notes Comput. Sci.*, vol. 3560, Springer Verlag, 2005, pp. 258–275.

[71] O. Gnawali, R. Fonseca, K. Jamieson, *et al.*, "CTP: An efficient, robust, and reliable collection tree protocol for wireless sensor networks," *ACM Trans. Sens. Networks*, vol. 10, no. 1, pp. 1–49, Nov. 2013.

[72] N. Burri, P. von Rickenbach, and R. Wattenhofer, "Dozer: Ultra-Low Power Data Gathering in Sensor Networks," in *2007 6th Int. Symp. Inf. Process. Sens. Networks*, IEEE, Apr. 2007, pp. 450–459.

[73] T. Winter, P. Thubert, A. Brandt, *et al.*, "Rpl: Ipv6 routing protocol for low-power and lossy networks," RFC Editor, RFC 6550, Mar. 2012, `http://www.rfc-editor.org/rfc/rfc6550.txt`.

[74] O. Iova, F. Theoleyre, and T. Noel, "Using multiparent routing in RPL to increase the stability and the lifetime of the network," *Ad Hoc Networks*, vol. 29, pp. 45–62, Jun. 2015.

[75] R.-A. Koutsiamanis, G. Z. Papadopoulos, X. Fafoutis, *et al.*, "From Best Effort to Deterministic Packet Delivery for Wireless Industrial IoT Networks," *IEEE Trans. Ind. Informatics*, vol. 14, no. 10, pp. 4468–4480, Oct. 2018.

[76] W. Tang, X. Ma, J. Huang, and J. Wei, "Toward Improved RPL: A Congestion Avoidance Multipath Routing Protocol with Time Factor for Wireless Sensor Networks," *J. Sensors*, vol. 2016, 2016.

[77] V. Loscri and S. Marano, "A new geographic multipath protocol for ad hoc networks to reduce the route coupling phenomenon," in *2006 IEEE 63rd Vehicular Technology Conference*, IEEE, vol. 3, 2006, pp. 1102–1106.

[78] T. Watteyne, A. Molinaro, M. G. Richichi, and M. Dohler, "From MANET To IETF ROLL Standardization: A Paradigm Shift in WSN Routing Protocols," *IEEE Commun. Surv. Tutorials*, vol. 13, no. 4, pp. 688–707, 2011.

[79] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[80] B. C. Villaverde, S. Rea, and D. Pesch, "Qos-aware routing for industrial wireless sensor networks," *Industrial Wireless Sensor Networks: Applications, Protocols, and Standards*, p. 259, 2017.

[81] J. Saleem, M. Hammoudeh, M. MHammoudeh, *et al.*, "IoT Standardisation - Challenges, Perspectives and Solution," vol. 9, 2018.

[82] L. Ladid, S. Östlund, and D. Führer, "D3.1: Identification of Standardisation Activities for IoT Technologies," EXCITING Consortium, Tech. Rep., 2018.

[83] "Ieee standard for telecommunications and information exchange between systems - lan/man specific requirements - part 15: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpan)," *IEEE Std 802.15.4-2003*, pp. 1–680, 2003.

[84] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans)," *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, pp. 1–320, Sep. 2006.

[85] "Ieee standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans)," *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)*, pp. 1–314, Sep. 2011.

[86] "Ieee standard for low-rate wireless networks," *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)*, pp. 1–709, Apr. 2016.

[87] "Ieee standard for low-rate wireless networks," *IEEE Std 802.15.4-2020 (Revision of IEEE Std 802.15.4-2015)*, pp. 1–800, 2020.

[90] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of ipv6 packets over ieee 802.15.4 networks,” RFC Editor, RFC 4944, Sep. 2007, http://www.rfc-editor.org/rfc/rfc4944.txt.

[91] J. Hui and P. Thubert, “Compression format for ipv6 datagrams over ieee 802.15.4-based networks,” RFC Editor, RFC 6282, Sep. 2011, http://www.rfc-editor.org/rfc/rfc6282.txt.

[92] Z. Shelby, S. Chakrabarti, E. Nordmark, and C. Bormann, “Neighbor discovery optimization for ipv6 over low-power wireless personal area networks (6lowpans),” RFC Editor, RFC 6775, Nov. 2012.

[94] Z. Shelby, K. Hartke, and C. Bormann, “The constrained application protocol (coap),” RFC Editor, RFC 7252, Jun. 2014, http://www.rfc-editor.org/rfc/rfc7252.txt.

[95] J. Postel, “User datagram protocol,” RFC Editor, STD 6, Aug. 1980, http://www.rfc-editor.org/rfc/rfc768.txt.

[96] G. Selander, J. Mattsson, F. Palombini, and L. Seitz, “Object security for constrained restful environments (oscore),” RFC Editor, RFC 8613, Jul. 2019.

[97] S. Deering and R. Hinden, “Internet protocol, version 6 (ipv6) specification,” RFC Editor, STD 86, Jul. 2017.

[98] P. Thubert, “Objective function zero for the routing protocol for low-power and lossy networks (rpl),” RFC Editor, RFC 6552, Mar. 2012, http://www.rfc-editor.org/rfc/rfc6552.txt.

[99] A. Conta, S. Deering, and M. Gupta, “Internet control message protocol (icmpv6) for the internet protocol version 6 (ipv6) specification,” RFC Editor, RFC 4443, Mar. 2006, http://www.rfc-editor.org/rfc/rfc4443.txt.

[100] Q. Wang, X. Vilajosana, and T. Watteyne, “6tisch operation sublayer (6top) protocol (6p),” RFC Editor, RFC 8480, Nov. 2018.

[101] X. Vilajosana, K. Pister, and T. Watteyne, “Minimal ipv6 over the tsch mode of ieee 802.15.4e (6tisch) configuration,” RFC Editor, BCP 210, May 2017.

[102] T. Chang, M. Vucinic, X. Vilajosana, S. Duquennoy, and D. Dujovne, “6tisch minimal scheduling function (msf),” IETF Secretariat, Internet-Draft draft-ietf-6tisch-msf-18, Sep. 2020, http://www.ietf.org/internet-drafts/draft-ietf-6tisch-msf-18.txt.

[103] F. Righetti, C. Vallati, D. Comola, and G. Anastasi, “Performance Measurements of IEEE 802.15.4g Wireless Networks,” in *2019 IEEE 20th Int. Symp. "A World Wireless, Mob. Multimed. Networks"*, IEEE, Jun. 2019, pp. 1–6.

[104] K. Yedavalli and B. Krishnamachari, “Enhancement of the IEEE 802.15.4 MAC protocol for scalable data collection in dense sensor networks,” in *2008 6th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wirel. Networks Work.*, IEEE, Apr. 2008, pp. 152–161.

[105] G. Anastasi, M. Conti, and M. Di Francesco, "A Comprehensive Analysis of the MAC Unreliability Problem in IEEE 802.15.4 Wireless Sensor Networks," *IEEE Trans. Ind. Informatics*, vol. 7, no. 1, pp. 52–65, Feb. 2011.

[106] D. De Guglielmo, S. Brienza, and G. Anastasi, "IEEE 802.15.4e: A survey," *Comput. Commun.*, vol. 88, pp. 1–24, Aug. 2016.

[107] K. Pister and L. Doherty, "Tsmp: Time synchronized mesh protocol," *IASTED Distributed Sensor Networks*, vol. 391, no. 398, p. 61, 2008.

[109] D. De Guglielmo, B. Al Nahas, S. Duquennoy, T. Voigt, and G. Anastasi, "Analysis and Experimental Evaluation of IEEE 802.15.4e TSCH CSMA-CA Algorithm," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1573–1588, Feb. 2017.

[110] X. Vilajosana, Q. Wang, F. Chraim, and T. Watteyne, "A Realistic Energy Consumption Model for TSCH Networks," vol. 14, no. 2, pp. 482–489, 2014.

[111] A. Elsts, S. Duquennoy, X. Fafoutis, *et al.*, "Microsecond-Accuracy Time Synchronization Using the IEEE 802.15.4 TSCH Protocol," in *2016 IEEE 41st Conf. Local Comput. Networks Work. (LCN Work.*, IEEE, Nov. 2016, pp. 156–164.

[112] G. Z. Papadopoulos, A. Mavromatis, X. Fafoutis, *et al.*, "Guard Time Optimisation for Energy Efficiency in IEEE 802.15.4-2015 TSCH Links," in *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 190, 2017, pp. 56–63.

[113] Q. Wang, X. Vilajosana, T. Watteyne, R. Sudhaakar, and P. Zand, "Transporting coap messages over ieee802.15.4e information elements," IETF Secretariat, Internet-Draft draft-wang-6tisch-6top-coapie-00, Jul. 2014, `http://www.ietf.org/internet-drafts/draft-wang-6tisch-6top-coapie-00.txt`.

[114] A. Karaagac, E. De Poorter, and J. Hoebeke, "In-Band Network Telemetry in Industrial Wireless Sensor Networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 1, pp. 517–531, Mar. 2020.

[115] A. Brandt, J. Buron, and G. Porcu, "Home automation routing requirements in low-power and lossy networks," RFC Editor, RFC 5826, Apr. 2010, `http://www.rfc-editor.org/rfc/rfc5826.txt`.

[116] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, "Building automation routing requirements in low-power and lossy networks," RFC Editor, RFC 5867, Jun. 2010, `http://www.rfc-editor.org/rfc/rfc5867.txt`.

[117] M. Dohler, T. Watteyne, T. Winter, and D. Barthel, "Routing requirements for urban low-power and lossy networks," RFC Editor, RFC 5548, May 2009.

[118] O. Gnawali and P. Levis, "The minimum rank with hysteresis objective function," RFC Editor, RFC 6719, Sep. 2012, `http://www.rfc-editor.org/rfc/rfc6719.txt`.

[119] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wirel. Networks*, vol. 11, no. 4, pp. 419–434, Jul. 2005.

[120] R. Teles Hermeto, A. Gallais, and F. Theoleyre, "Scheduling for IEEE802.15.4-TSCH and slow channel hopping MAC in low power industrial wireless networks: A survey," *Comput. Commun.*, vol. 114, pp. 84–105, Dec. 2017.

[121] A. Elsts, S. Kim, H. S. Kim, and C. Kim, "An empirical survey of autonomous scheduling methods for TSCH," *IEEE Access*, vol. 8, pp. 67 147–67 165, 2020.

[122] F. Righetti, C. Vallati, S. K. Das, and G. Anastasi, "Analysis of Distributed and Autonomous Scheduling Functions for 6TiSCH Networks," *IEEE Access*, vol. 8, pp. 158 243–158 262, 2020.

[123] M. Vucinic, J. Simon, K. Pister, and M. Richardson, "Constrained join protocol (cojp) for 6tisch," IETF Secretariat, Internet-Draft draft-ietf-6tisch-minimal-security-15, Dec. 2019, `http://www.ietf.org/internet-drafts/draft-ietf-6tisch-minimal-security-15.txt`.

[124] M. Vucinic, B. Tourancheau, F. Rousseau, *et al.*, "OSCAR: Object security architecture for the Internet of Things," in *Proceeding IEEE Int. Symp. a World Wireless, Mob. Multimed. Networks 2014*, IEEE, Jun. 2014, pp. 1–10.

[125] M. Vucinic, B. Tourancheau, T. Watteyne, *et al.*, "DTLS performance in duty-cycled networks," in *2015 IEEE 26th Annu. Int. Symp. Pers. Indoor, Mob. Radio Commun.*, vol. 2015-Decem, IEEE, Aug. 2015, pp. 1333–1338. arXiv: `1507.05810`.

[126] S. Sciancalepore, M. Vučinić, G. Piro, G. Boggia, and T. Watteyne, "Link-layer security in TSCH networks: effect on slot duration," *Trans. Emerg. Telecommun. Technol.*, vol. 28, no. 1, e3089, Jan. 2017.

[127] T. Watteyne, V. Handziski, X. Vilajosana, *et al.*, "Industrial Wireless IP-Based Cyber –Physical Systems," *Proc. IEEE*, vol. 104, no. 5, pp. 1025–1038, May 2016.

[128] F. Battaglia, M. Collotta, L. Leonardi, L. Lo Bello, and G. Patti, "electronics Article,"

[129] F. Kauer, M. Köstler, and V. Turau, "Reliable Wireless Multi-Hop Networks with Decentralized Slot Management: An Analysis of IEEE 802.15.4 DSME," Jun. 2018. arXiv: `1806.10521`.

[130] H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "DynaMO - Dynamically tuning DSME Networks," *ACM SIGBED Rev.*, vol. 16, no. 4, pp. 8–13, Jan. 2020.

[132] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 15.1a: Wireless medium access control (mac) and physical layer (phy) specifications for wireless personal area networks (wpan)," *IEEE Std 802.15.1-2005 (Revision of IEEE Std 802.15.1-2002)*, pp. 1–700, 2005.

[136] M. Siekkinen, M. Hiienkari, J. K. Nurminen, and J. Nieminen, "How low energy is bluetooth low energy? Comparative measurements with ZigBee/802.15.4," in *2012 IEEE Wirel. Commun. Netw. Conf. Work. WCNCW 2012*, 2012, pp. 232–237.

[137] K. Mikhaylov, N. Plevritakis, and J. Tervonen, "Performance Analysis and Comparison of Bluetooth Low Energy with IEEE 802.15.4 and SimpliciTI," *J. Sens. Actuator Networks*, vol. 2, no. 3, pp. 589–613, Aug. 2013.

[138] P. Narendra, S. Duquennoy, and T. Voigt, "BLE and IEEE 802.15.4 in the IoT: Evaluation and Interoperability Considerations," in, 2016, pp. 427–438.

[139] G. Patti, L. Leonardi, and L. Lo Bello, "A Bluetooth Low Energy real-time protocol for Industrial Wireless mesh Networks," in *IECON 2016 - 42nd Annu. Conf. IEEE Ind. Electron. Soc.*, IEEE, Oct. 2016, pp. 4627–4632.

[140] R. Rondón, M. Gidlund, and K. Landernäs, "Evaluating Bluetooth Low Energy Suitability for Time-Critical Industrial IoT Applications," *Int. J. Wirel. Inf. Networks*, vol. 24, no. 3, pp. 278–290, Sep. 2017.

[141] M. Spörk, C. A. Boano, M. Zimmerling, and K. Römer, "BLEach," in *Proc. 15th ACM Conf. Embed. Netw. Sens. Syst.*, vol. 2017-Janua, New York, NY, USA: ACM, Nov. 2017, pp. 1–14.

[142] M. Spörk, C. A. Boano, and K. Römer, "Improving the Timeliness of Bluetooth Low Energy in Dynamic RF Environments," *ACM Trans. Internet Things*, vol. 1, no. 2, pp. 1–32, Apr. 2020.

[144] C. Gomez, S. Darroudi, T. Savolainen, and M. Spoerk, "Ipv6 mesh over bluetooth(r) low energy using ipsp," IETF Secretariat, Internet-Draft draft-ietf-6lo-blemesh-09, Dec. 2020, `http://www.ietf.org/internet-drafts/draft-ietf-6lo-blemesh-09.txt`.

[145] S. M. Darroudi, C. Gomez, and J. Crowcroft, "Bluetooth Low Energy Mesh Networks: A Standards Perspective," *IEEE Commun. Mag.*, vol. 58, no. 4, pp. 95–101, Apr. 2020.

[147] "Ieee standard for information technology–telecommunications and information exchange between systems - local and metropolitan area networks–specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 2: Sub 1 ghz license exempt operation," *IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as amended by IEEE Std 802.11ai-2016)*, pp. 1–594, 2017.

[148] G. Tian, S. Camtepe, and Y. C. Tian, "A Deadline-Constrained 802.11 MAC Protocol with QoS Differentiation for Soft Real-Time Control," *IEEE Trans. Ind. Informatics*, vol. 12, no. 2, pp. 544–554, Apr. 2016.

[149] L. F. Del Carpio, P. Di Marco, P. Skillermark, R. Chirikov, and K. Lagergren, "Comparison of 802.11ah, BLE and 802.15.4 for a Home Automation Use Case," *Int. J. Wirel. Inf. Networks*, vol. 24, no. 3, pp. 243–253, 2017.

[150] L. Polak, P. Jurak, and J. Milos, "MATLAB-Based PHY Simulators for Performance Study of the IEEE 802.11ah/af Systems," in *2020 43rd Int. Conf. Telecommun. Signal Process.*, IEEE, Jul. 2020, pp. 184–187.

[157] *Industrial networks - wireless communication network and communication profiles - wirelesshart*, 3, rue de Varembé, Geneva: IEC, 2016.

[158] *Wireless systems for industrial automation: Process control and related applications*, 67 TW Alexander Dr, Durham, NC: ANSI/ISA, 2011.

[159] *Industrial networks – wireless communication network and communication profiles – wia-pa*, 3, rue de Varembé, Geneva: IEC, 2015.

[160] T. Lennvall, K. Landernäs, M. Gidlund, and J. Åkerberg, "Industrial wsn standards," 2013.

[161] S. Petersen and S. Carlsen, "WirelessHART Versus ISA100.11a: The Format War Hits the Factory Floor," *IEEE Ind. Electron. Mag.*, vol. 5, no. 4, pp. 23–34, Dec. 2011.

[162] A. Koubâa, A. Cunha, M. Alves, and E. Tovar, "TDBS: a time division beacon scheduling mechanism for ZigBee cluster-tree wireless sensor networks," *Real-Time Syst.*, vol. 40, no. 3, pp. 321–354, Dec. 2008.

[163] E. Toscano and L. L. Bello, "Multichannel Superframe Scheduling for IEEE 802.15.4 Industrial Wireless Sensor Networks," *IEEE Trans. Ind. Informatics*, vol. 8, no. 2, pp. 337–350, May 2012.

[164] P. Muthukumaran, R. De Paz, R. Spinar, and D. Pesch, "MeshMAC: Enabling mesh networking over IEEE 802.15.4 through distributed beacon scheduling," in *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng.*, vol. 28 LNICST, Springer, Berlin, Heidelberg, 2010, pp. 561–575.

[165] B. C. Villaverde, R. De Paz Alberola, S. Rea, and D. Pesch, "Experimental Evaluation of Beacon Scheduling Mechanisms for Multihop IEEE 802.15.4 Wireless Sensor Networks," in *2010 Fourth Int. Conf. Sens. Technol. Appl.*, IEEE, Jul. 2010, pp. 226–231.

[166] W.-y. Lee, K.-i. Hwang, Y.-A. Jeon, and S. Choi, "Distributed Fast Beacon Scheduling for Mesh Networks," in *2011 IEEE Eighth Int. Conf. Mob. Ad-Hoc Sens. Syst.*, IEEE, Oct. 2011, pp. 727–732.

[167] K.-i. Hwang and S.-w. Nam, "Analysis and Enhancement of IEEE 802.15.4e DSME Beacon Scheduling Model," *J. Appl. Math.*, vol. 2014, pp. 1–15, 2014.

[168] C. Cano and D. Malone, "A learning approach to decentralised beacon scheduling," *Ad Hoc Networks*, vol. 49, pp. 58–69, Oct. 2016.

[169] D. De Guglielmo, A. Seghetti, G. Anastasi, and M. Conti, "A performance analysis of the network formation process in IEEE 802.15.4e TSCH wireless sensor/actuator networks," in *2014 IEEE Symp. Comput. Commun.*, IEEE, Jun. 2014, pp. 1–6.

[170] T. P. Duy, T. Dinh, and Y. Kim, "A rapid joining scheme based on fuzzy logic for highly dynamic IEEE 802.15.4e time-slotted channel hopping networks," *Int. J. Distrib. Sens. Networks*, vol. 12, no. 8, p. 155 014 771 665 942, Aug. 2016.

[171] A. Kalita and M. Khatua, "Channel Condition Based Dynamic Beacon Interval for Faster Formation of 6TiSCH Network," *IEEE Trans. Mob. Comput.*, vol. 1233, no. c, pp. 1–1, 2020.

[172] J. Vera-Pérez, D. Todolí-Ferrandis, J. Silvestre-Blanes, and V. Sempere-Payá, "Bell-X, An Opportunistic Time Synchronization Mechanism for Scheduled Wireless Sensor Networks," *Sensors*, vol. 19, no. 19, p. 4128, Sep. 2019.

[174] E. Vogli, G. Ribezzo, L. A. Grieco, and G. Boggia, "Fast join and synchronization schema in the IEEE 802.15.4e MAC," in *2015 IEEE Wirel. Commun. Netw. Conf. Work.*, IEEE, Mar. 2015, pp. 85–90.

[175] D. De Guglielmo, S. Brienza, and G. Anastasi, "A Model-based Beacon Scheduling algorithm for IEEE 802.15.4e TSCH networks," in *2016 IEEE 17th Int. Symp. A World Wireless, Mob. Multimed. Networks*, IEEE, Jun. 2016, pp. 1–9.

[176] I. Khoufi and P. Minet, "An enhanced deterministic beacon advertising algorithm for building TSCH networks," *Ann. Telecommun.*, vol. 73, no. 11-12, pp. 745–757, Dec. 2018.

[177] A. Karalis, D. Zorbas, and C. Douligeris, "Collision-Free Broadcast Methods for IEEE 802.15.4-TSCH Networks Formation," in *Proc. 21st ACM Int. Conf. Model. Anal. Simul. Wirel. Mob. Syst.*, New York, NY, USA: ACM, Oct. 2018, pp. 91–98.

[178] J.-Y. Kim, S.-H. Chung, and Y.-V. Ha, "A fast joining scheme based on channel quality for IEEE802.15.4e TSCH in severe interference environment," in *2017 Ninth Int. Conf. Ubiquitous Futur. Networks*, IEEE, Jul. 2017, pp. 427–432.

[179] N. Accettura, L. A. Grieco, G. Boggia, and P. Camarda, "Performance analysis of the RPL Routing Protocol," *2011 IEEE Int. Conf. Mechatronics, ICM 2011 - Proc.*, pp. 767–772, 2011.

[180] H. Kermajani and C. Gomez, "On the network convergence process in RPL over IEEE 802.15.4 multihop networks: Improvement and trade-offs," *Sensors (Switzerland)*, vol. 14, no. 7, pp. 11 993–12 022, 2014.

[182] B. Pavkovic, A. Duda, W.-j. Hwang, and F. Theoleyre, "Efficient topology construction for RPL over IEEE 802.15.4 in wireless sensor networks," *Ad Hoc Networks*, vol. 15, pp. 25–38, Apr. 2014.

[183] M. Vucinic, G. Romaniello, L. Guelorget, *et al.*, "Topology construction in RPL networks over beacon-enabled 802.15.4," in *2014 IEEE Symp. Comput. Commun.*, IEEE, Jun. 2014, pp. 1–7. arXiv: 1404.7803.

[184] O. Iova, F. Theoleyre, T. Watteyne, and T. Noel, "The Love-Hate Relationship between IEEE 802.15.4 and RPL," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 188–194, Jan. 2017.

[185] S. Duquennoy, B. Al Nahas, O. Landsiedel, and T. Watteyne, "Orchestra," in *Proc. 13th ACM Conf. Embed. Networked Sens. Syst. - SenSys '15*, New York, New York, USA: ACM Press, 2015, pp. 337–350.

[186] T. Chang, M. Vučinić, X. V. Guillén, D. Dujovne, and T. Watteyne, "<scp>6TiSCH</scp> minimal scheduling function: Performance evaluation," *Internet Technol. Lett.*, vol. 3, no. 4, pp. 4–9, Jul. 2020.

[187] L. Wang and A. Reinhardt, "A simulative study of network association delays in IEEE 802.15.4e TSCH networks," in *18th IEEE Int. Symp. A World Wireless, Mob. Multimed. Networks, WoWMoM 2017 - Conf.*, Institute of Electrical and Electronics Engineers Inc., Jul. 2017.

[188] F. Righetti, C. Vallati, G. Anastasi, and S. Das, "Performance Evaluation the 6top Protocol and Analysis of its Interplay with Routing," in *2017 IEEE Int. Conf. Smart Comput.*, IEEE, May 2017, pp. 1–6.

[189] M. Vučinić, T. Watteyne, and X. Vilajosana, "Broadcasting strategies in 6TiSCH networks," *Internet Technol. Lett.*, vol. 1, no. 1, e15, Jan. 2018.

[190] J. Vera-Pérez, D. Todolí-Ferrandis, S. Santonja-Climent, J. Silvestre-Blanes, and V. Sempere-Payá, "A Joining Procedure and Synchronization for TSCH-RPL Wireless Sensor Networks," *Sensors*, vol. 18, no. 10, p. 3556, Oct. 2018.

[191] C. Vallati, S. Brienza, G. Anastasi, and S. K. Das, "Improving Network Formation in 6TiSCH Networks," *IEEE Trans. Mob. Comput.*, vol. 18, no. 1, pp. 98–110, Jan. 2019.

[192] W.-B. Pöttner, H. Seidel, J. Brown, U. Roedig, and L. Wolf, "Constructing Schedules for Time-Critical Data Delivery in Wireless Sensor Networks," *ACM Trans. Sens. Networks*, vol. 10, no. 3, pp. 1–31, Apr. 2014.

[193] G. Gaillard, D. Barthel, F. Theoleyre, and F. Valois, "Kausa: KPI-aware Scheduling Algorithm for Multi-flow in Multi-hop IoT Networks," in *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9724, Springer Verlag, 2016, pp. 47–61.

[194] P. Minet and Y. Tanaka, "Optimal Number of Message Transmissions forProbabilistic Guarantee of Latency in the IoT," *Sensors*, vol. 19, no. 18, p. 3970, Sep. 2019.

[195] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proc. first Int. Conf. Embed. networked Sens. Syst. - SenSys '03*, New York, New York, USA: ACM Press, 2003, p. 14.

[196] M. Senel, K. Chintalapudi, D. Lal, A. Keshavarzian, and E. J. Coyle, "A Kalman Filter Based Link Quality Estimation Scheme for Wireless Sensor Networks," in *IEEE GLOBECOM 2007-2007 IEEE Glob. Telecommun. Conf.*, IEEE, Nov. 2007, pp. 875–880.

[197] S. Duquennoy, J. Eriksson, and T. Voigt, "Five-Nines Reliable Downward Routing in RPL," no. October, Oct. 2017. arXiv: 1710.02324.

[198] O. Iova, F. Theoleyre, and T. Noel, "Stability and efficiency of RPL under realistic conditions in Wireless Sensor Networks," *IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. PIMRC*, pp. 2098–2102, 2013.

[199] N. Baccour, A. Koubâa, H. Youssef, *et al.*, "F-LQE: A Fuzzy Link Quality Estimator for Wireless Sensor Networks," in *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5970 LNCS, Springer, Berlin, Heidelberg, 2010, pp. 240–255.

[200] S. Rekik, N. Baccour, M. Jmaiel, and K. Drira, "Low-Power link quality estimation in smart grid environments," in *2015 Int. Wirel. Commun. Mob. Comput. Conf.*, IEEE, Aug. 2015, pp. 1211–1216.

[201] ——, "Holistic link quality estimation-based routing metric for RPL networks in smart grids," in *2016 IEEE 27th Annu. Int. Symp. Pers. Indoor, Mob. Radio Commun.*, IEEE, Sep. 2016, pp. 1–6.

[202] C. Vallati, E. Ancillotti, R. Bruno, E. Mingozzi, and G. Anastasi, "Interplay of Link Quality Estimation and RPL Performance," in *Proc. 13th ACM Symp. Perform. Eval. Wirel. Ad Hoc, Sensor, Ubiquitous Networks - PE-WASUN '16*, New York, New York, USA: ACM Press, 2016, pp. 83–90.

[203] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, "Four-Bit Wireless Link Estimation," in *HotNets*, 2007.

[204] S. Munir, S. Lin, E. Hoque, *et al.*, "Addressing burstiness for reliable communication and latency bound generation in wireless sensor networks," in *Proc. 9th ACM/IEEE Int. Conf. Inf. Process. Sens. Networks - IPSN '10*, New York, New York, USA: ACM Press, 2010, p. 303.

[205] J. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing metrics used for path calculation in low-power and lossy networks," RFC Editor, RFC 6551, Mar. 2012, `http://www.rfc-editor.org/rfc/rfc6551.txt`.

[206] P. Ruckebusch, J. Devloo, D. Carels, E. De Poorter, and I. Moerman, "An Evaluation of Link Estimation Algorithms for RPL in Dynamic Wireless Sensor Networks," in *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 170, 2016, pp. 349–361.

[207] S. Dawans, S. Duquennoy, and O. Bonaventure, "On link estimation in dense RPL deployments," in *37th Annu. IEEE Conf. Local Comput. Networks – Work.*, IEEE, Oct. 2012, pp. 952–955.

[208] J. Kantert, F. Reinhard, G. v. Zengen, *et al.*, "Combining trust and etx to provide robust wireless sensor networks," in *ARCS 2016; 29th International Conference on Architecture of Computing Systems*, 2016, pp. 1–7.

[209] D. T. Delaney, L. Xu, and G. M. P. O'Hare, "Spreading the Load in a Tree Type Routing Structure," in *2013 22nd Int. Conf. Comput. Commun. Networks*, IEEE, Jul. 2013, pp. 1–7.

[210] E. Ancillotti, R. Bruno, M. Conti, E. Mingozzi, and C. Vallati, "Trickle-L^2: Lightweight link quality estimation through Trickle in RPL networks," in *Proceeding IEEE Int. Symp. a World Wireless, Mob. Multimed. Networks 2014*, IEEE, Jun. 2014, pp. 1–9.

[211] D. Liu, Z. Cao, Y. Zhang, and M. Hou, "Achieving accurate and real-time link estimation for low power wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2096–2109, Aug. 2017.

[212] E. Ancillotti, R. Bruno, and M. Conti, "Reliable Data Delivery With the IETF Routing Protocol for Low-Power and Lossy Networks," *IEEE Trans. Ind. Informatics*, vol. 10, no. 3, pp. 1864–1877, Aug. 2014.

[213] L. Pradittasnee, S. Camtepe, and Y. C. Tian, "Efficient Route Update and Maintenance for Reliable Routing in Large-Scale Sensor Networks," *IEEE Trans. Ind. Informatics*, vol. 13, no. 1, pp. 144–156, Feb. 2017.

[214] E. Ancillotti, C. Vallati, R. Bruno, and E. Mingozzi, "A reinforcement learning-based link quality estimation strategy for RPL and its impact on topology management," *Comput. Commun.*, vol. 112, pp. 1–13, 2017.

[215] R. D. Gomes, D. V. Queiroz, A. C. Lima Filho, I. E. Fonseca, and M. S. Alencar, "Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes," *Ad Hoc Networks*, vol. 59, pp. 116–133, May 2017.

[216] C. Vallati and E. Mingozzi, "Trickle-F: Fair broadcast suppression to improve energy-efficient route formation with the RPL routing protocol," in *2013 Sustain. Internet ICT Sustain.*, IEEE, Oct. 2013, pp. 1–9.

[217] R. Teles Hermeto, A. Gallais, K. V. Laerhoven, and F. Theoleyre, "Passive Link Quality Estimation for Accurate and Stable Parent Selection in Dense 6TiSCH Networks," *Int. Conf. Embed. Wirel. Syst. Networks 2018*, pp. 114–125, 2018.

[218] P. H. Gomes and B. Krishnamachari, "TAMU-RPL: Thompson sampling-based multichannel RPL," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 2, pp. 1–20, Feb. 2020.

[219] A. Elsts, X. Fafoutis, R. Piechocki, and I. Craddock, "Adaptive channel selection in IEEE 802.15.4 TSCH networks," in *2017 Glob. Internet Things Summit*, IEEE, Jun. 2017, pp. 1–6.

[220] V. Kotsiou, G. Z. Papadopoulos, P. Chatzimisios, and F. Theoleyre, "LABeL," in *Proc. 20th ACM Int. Conf. Model. Anal. Simul. Wirel. Mob. Syst.*, vol. 2017-Novem, New York, NY, USA: ACM, Nov. 2017, pp. 25–33.

[221] B. Kerkez, T. Watteyne, M. Magliocco, S. Glaser, and K. Pister, "Feasibility analysis of controller design for adaptive channel hopping," in *Proc. 4th Int. ICST Conf. Perform. Eval. Methodol. Tools*, ICST, 2009.

[222] P. Du and G. Roussos, "Adaptive time slotted channel hopping for wireless sensor networks," in *2012 4th Comput. Sci. Electron. Eng. Conf.*, IEEE, Sep. 2012, pp. 29–34.

[223] R. Tavakoli, M. Nabi, T. Basten, and K. Goossens, "Enhanced Time-Slotted Channel Hopping in WSNs Using Non-intrusive Channel-Quality Estimation," in *2015 IEEE 12th Int. Conf. Mob. Ad Hoc Sens. Syst.*, IEEE, Oct. 2015, pp. 217–225.

[224] Peishuo Li, T. Vermeulen, H. Liy, and S. Pollin, "An adaptive channel selection scheme for reliable TSCH-based communication," in *2015 Int. Symp. Wirel. Commun. Syst.*, vol. 2016-April, IEEE, Aug. 2015, pp. 511–515.

[225] P. H. Gomes, T. Watteyne, and B. Krishnamachari, "MABO-TSCH: Multihop and blacklist-based optimized time synchronized channel hopping," *Trans. Emerg. Telecommun. Technol.*, vol. 29, no. 7, e3223, Jul. 2018.

[226] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[227] C. Vallati, S. Brienza, G. Anastasi, and S. K. Das, "Improving Network Formation in 6TiSCH Networks," *IEEE Trans. Mob. Comput.*, vol. 18, no. 1, pp. 98–110, Jan. 2019.

[228] N. Accettura, E. Vogli, M. R. Palattella, *et al.*, "Decentralized Traffic Aware Scheduling in 6TiSCH Networks: Design and Experimental Evaluation," *IEEE Internet Things J.*, vol. 2, no. 6, pp. 455–470, Dec. 2015.

[229] P. Levis, T. H. Clausen, J. W. Hui, O. Gnawali, and J. Ko, "The trickle algorithm," RFC Editor, RFC 6206, Mar. 2011, `http://www.rfc-editor.org/rfc/rfc6206.txt`.

[230] M. R. Palattella, X. Vilajosana, T. Chang, M. A. R. Ortega, and T. Watteyne, "Lessons learned from the 6TiSCH plugtests," in *Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng. LNICST*, vol. 170, Springer Verlag, Oct. 2016, pp. 415–426.

[232] F. Righetti, C. Vallati, G. Anastasi, and S. K. Das, “Analysis and Improvement of the On-The-Fly Bandwidth Reservation Algorithm for 6TiSCH,” in *19th IEEE Int. Symp. a World Wireless, Mob. Multimed. Networks, WoWMoM 2018*, IEEE, Jun. 2018, pp. 1–9.

[233] D. Fanucchi, B. Staehle, and R. Knorr, “Network Formation for Industrial IoT: Evaluation, Limits and Recommendations,” in *2018 IEEE 23rd Int. Conf. Emerg. Technol. Fact. Autom.*, vol. 2018-Septe, IEEE, Sep. 2018, pp. 227–234.

[234] T. Clausen, U. Herberg, and M. Philipp, “A critical evaluation of the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL),” *Int. Conf. Wirel. Mob. Comput. Netw. Commun.*, pp. 365–372, 2011.

[235] T. Istomin, O. Iova, G. P. Picco, and C. Kiraly, “Route or flood? Reliable and efficient support for downward traffic in RPL,” *ACM Trans. Sens. Networks*, vol. 16, no. 1, 2019.

Websites

[3] IEEEE Standards Association (IEEE-SA). (2021). Homepage, [Online]. Available: `https://standards.ieee.org/`.

[5] International Society of Automation (ISA). (2021). Setting the Standard for Automation, [Online]. Available: `https://www.isa.org/`.

[8] VDE. (May 2021). Positionspapier "Funktechnologien für Industrie 4.0", [Online]. Available: `https://shop.vde.com/en/vde-positionspapier-funktechnologien-fuer-industrie-40`.

[10] Contiki Community. (2004). Homepage, [Online]. Available: `http://www.contiki-os.org/`.

[11] Contiki-NG. (Nov. 2017). Homepage, [Online]. Available: `https://www.contiki-ng.org/`.

[12] Zolertia Inc. (2017). Re-mote platform wiki-page, [Online]. Available: `https://github.com/Zolertia/Resources/wiki/RE-M`.

[14] Internet Engineering Task Force (IETF). (2021). Homepage, [Online]. Available: `https://www.ietf.org/`.

[18] Energizer. (2021). Ultimate lithium - product datasheet, [Online]. Available: `https://data.energizer.com/PDFs/l91.pdf`.

[22] Industrial Shields. (2016). Open mote b - product description, [Online]. Available: `https://www.industrialshields.com/open-mote-b-industrial-shields-open-source-device-ready-for-internet-of-things`.

[23] Crossbow. (2005). Micaz mote platform - product description, [Online]. Available: `https://www.willow.co.uk/MICAz_OEM_Edition_Datasheet.pdf`.

[24] ——, (2004). Telosb mote platform - product description, [Online]. Available: https://www.willow.co.uk/TelosB_Datasheet.pdf.

[25] Zolertia Inc. (2018). Zolertia z1 mote platform wiki-page, [Online]. Available: https://github.com/Zolertia/Resources/wiki/The-Z1-mote.

[26] Linear Technology Corporation. (2016). Smartmesh ip access point mote - product description, [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/59012ipaf.pdf.

[32] TinyOS Community. (2000). Homepage, [Online]. Available: http://www.tinyos.net/.

[33] RIOT Community. (2013). Homepage, [Online]. Available: https://www.riot-os.org/.

[34] FreeRTOS Community. (2003). Homepage, [Online]. Available: https://www.freertos.org/.

[35] Silicon Laboratories. (2021). $\mu C/$ OS RTOS and Stacks, [Online]. Available: https://www.micrium.com/rtos/.

[36] A. Dunkels. (Jan. 2002). Uip - a free small tcp/ip stack, [Online]. Available: http://www.dunkels.com/adam/download/uip-doc-0.6.pdf.

[38] B. Thebaudeau. (2019). Information about contiki mac protocols, [Online]. Available: https://github.com/contiki-os/contiki/wiki/Change-mac-or-radio-duty-cycling-protocols.

[88] S. Olshansky. (Sep. 2019). Ietf 105: Internet of things wrapup, [Online]. Available: https://www.ietf.org/blog/ietf105-iot-wrapup/.

[89] IEEE. (2021). Ieee 802.15 working group for wireless specialty networks, [Online]. Available: http://ieee802.org/15/.

[93] IETF. (Sep. 2013). Ipv6 over the tsch mode of ieee 802.15.4e (6tisch), [Online]. Available: https://datatracker.ietf.org/wg/6tisch/about/.

[108] FieldComm Group. (2021). HART Digital Communication Technologiy in the Process Industries, [Online]. Available: https://www.fieldcommgroup.org/technologies/hart.

[131] M. Köstler. (2020). OpenDSME, Open-Source implementation of the IEEE 802.15.4 DSME Link Layer, [Online]. Available: http://opendsme.org/.

[133] Texas Instruments. (2021). CC2650 data sheet, product information and support, [Online]. Available: https://www.ti.com/product/CC2650.

[134] Nordic Semiconductor. (2021). nRF52840 - Bluetooth 5.2 SoC, [Online]. Available: https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF52840.

[135] Microchip Technology Inc. (2021). AT86RF215 - Wireless Modules, [Online]. Available: https://www.microchip.com/wwwproducts/en/AT86RF215.

[143] Bluetooth SIG. (2021). Specifications List for Mesh Topologies, [Online]. Available: https://www.bluetooth.com/specifications/specs/?status=all&keyword=mesh&filter=.

[146] IETF. (2013). Ipv6 over networks of resource-constrained nodes (6lo), [Online]. Available: https://datatracker.ietf.org/wg/6lo/about/.

[151] 3GPP. (2021). 3gpp website, [Online]. Available: https://www.3gpp.org/.

[152] ——, (Sep. 2015). Introduction of Release 13 NB-IoT feature to LTE RAN1 specifications, [Online]. Available: https://portal.3gpp.org/ngppapp/CreateTdoc.aspx?mode=view&contributionUid=RP-161067.

[153] ——, (2019). Release 15, [Online]. Available: https://www.3gpp.org/release-15.

[154] LoRa Alliance. (2021). Website, [Online]. Available: https://lora-alliance.org/.

[155] ——, (2021). LoRaWAN® 1.0.4 Specification Package, [Online]. Available: https://lora-alliance.org/resource_hub/lorawan-104-specification-package/.

[156] ——, (2021). Technical Document about LoRa® Modulation, [Online]. Available: https://lora-developers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/.

[173] A. Elsts. (Sep. 2020). Documentation of tsch and 6tisch in contiki-ng, [Online]. Available: https://github.com/contiki-ng/contiki-ng/wiki/Documentation:-TSCH-and-6TiSCH.

[181] OMNeT++ Community. (2020). Homepage, [Online]. Available: https://omnetpp.org/.

[231] Texas Instruments. (2013). Cc2538 powerful wireless mcu - datasheet, [Online]. Available: https://www.ti.com/product/CC2538.