KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Technology

Use Case Adaptive Multi-Mode Platform for Energy Efficient Devices in the Internet of Things

Guus Leenders

Supervisors:
Prof. dr. ir. L. De Strycker
Prof. dr. ir. L. Van der Perre

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Technology (PhD)

November 2023

Use Case Adaptive Multi-Mode Platform for Energy Efficient Devices in the Internet of Things

Guus LEENDERS

Examination committee:
Prof. dr. ir. M. Vergauwen, chair
Prof. dr. ir. L. De Strycker, supervisor
Prof. dr. ir. L. Van der Perre, supervisor
Prof. dr. ir. P. Vandewalle
Dr. ir. H. Sallouha
Ing. B. Pen
(Tesla Networks, Belgium)
Prof. dr. ir. J. Famaey
(University of Antwerp)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Technology (PhD)

November 2023

Uitgegeven in eigen beheer, Guus Leenders, Gebroeders De Smetstraat 1, B-9000 Ghent (Belgium)

Preface

A single man cannot build a house.

(Nigerian parable)

The pages that follow are the culmination of four years of research, countless hours of contemplation, and, most significantly, numerous and never ending collaborations. None of the projects showcased in this thesis are solely my doing, in this section I wish to explicitly credit all those who have been involved.

In 2015, I joined the Dramco research group with a firm conviction that a PhD was not in my future as I was uncertain about the subject worthy of a four-year exploration. Engaged in diverse company projects facilitated by Jean Pierre's invaluable insights, I found a common thread in the most enjoyable endeavors: the integration of "smart" elements in day-to-day objects – be it in street lighting, indoor lighting, mattresses, and more. The prospect of understanding the energy implications of appending the term "smart" intrigued me deeply: How much would energy consumption escalate? This curiosity became the focal point of my research, a subject I enthusiastically delved into over the course of four years.

I extend my gratitude to Jean Pierre for providing me with an initial exposure to research across a diverse array of projects. Special thanks to Lieven and Liesbet for affording me the opportunity to embark on a PhD journey in a subject that truly ignites my passion. Lieven, your invaluable guidance and the readiness to contribute timely and insightful perspectives have been instrumental. Liesbet, your consistent feedback and unwavering support have been indispensable.

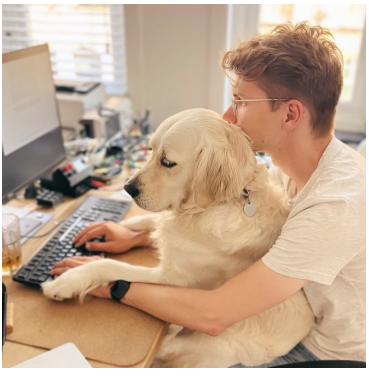
My colleagues at Dramco have played a pivotal role in bringing this Ph.D. to fruition. Gilles, your consistent collaboration on both software and academic fronts has been indispensable. You epitomize the essence of a true academic – a standard that any academic should aspire to reach. Geoffrey, your enthusiastic partnership in exploring our "smart" ideas has been invaluable. Thank you for your mentorship,

particularly relating to all things C. Stijn, our stimulating discussions in B221, occasionally audible in the corridors, have been a source of inspiration. Bart, I appreciate your guidance in mastering the intricacies of developing electronic hardware, turning it into a true art. Jarne, thank you for sharing my passion for hardware development and matching my enthusiasm for new projects. Chesney, your unwavering attention when I needed to pick someone else's brain, unrelated to IoT, has been a tremendous support. To Alex, Bert, Carine, Daan, Emanuele, Fan, François, Hugo, Jona, Jorik, Karel, Kevin, Laura, Matthias, Nobby, Qingli, Sam, Sara, Sarah, Thomas, Willem, and Weronika – thank you for collectively creating the home that Dramco represents.

This PhD has been shaped under the guidance of the supervisory committee, and I wish to extend my appreciation to my supervisors and assessors: Prof. Lieven De Strycker, Prof. Liesbet Van der Perre, Prof. Patrick Vandewalle, and Ing. Bart Pen, for their constructive remarks and invaluable advice. My gratitude also goes to the additional members of the examination jury, Prof. Jeroen Famaey and Dr. Hazem Sallouha, for their insightful feedback that enhanced the quality of this work. Special thanks to Prof. Maarten Vergauwen for presiding the jury.

Finally, I want to express my sincere appreciation to my family and friends for their steadfast support and genuine interest in my ramblings about low-power IoT topics. A special acknowledgment is due to Niels, my parents, and Rien for their confidence and constant moral support.

Guus Leenders
November 10, 2023



Abstract

The widespread adoption of Internet of Things (IoT) devices has led to an increasing need for efficient and sustainable operation of low-power IoT nodes. A crucial factor in achieving low-power operation is optimizing one of the highest energy spenders in an Internet of Things (IoT) node: a node's wireless communication capabilities. Various new network technologies, known as low power wide area networks (LPWANs), have been developed to provide low bandwidth, long range, and low power communication. However, no single network can be considered the ultimate solution for all IoT devices. This work aims to characterize these networks with a strong focus on energy expenditure and explore the advantages of combining multiple IoT networks on a single IoT node.

The first phase of the study involves characterizing the energy consumption profiles of individual radio access technologies (RATs) under realistic conditions, including aspects such as hardware development and network deployments. Factors such as data transmission rates, coverage distance, and packet latency are examined to understand their impact on a node's autonomy.

Based on the findings from the single-RAT analysis, this research investigates the benefits and challenges of implementing multi-RAT approaches for IoT node communication: combining two or more wireless IoT technologies. Two approaches are considered: integrating multiple RATs into a single IoT node and building multi-RAT into networks of IoT nodes.

By integrating complementary wireless technologies within a single node, the ability to select the most suitable technology based on specific circumstances and requirements brings significant advantages, including not only energy savings but also enhanced coverage and improved quality of service (QoS).

Complementary wireless technologies can also be deployed and shared across a larger IoT network, enabling a distributed approach. By doing so, only a few nodes in the network need to be equipped with a cloud connection, and consequently

serve as central data sinks. To facilitate this, a network protocol is proposed to transmit data to the central data sinks, which not only ensures low-power operation for all nodes in the network, but also increases the network coverage area via a multi-hop strategy for all nodes.

The research outcomes are applied to various use cases through a case study approach, focusing on IoT monitoring scenarios such as sound disturbance monitoring in smart cities and monitoring tree health and wildfires in smart forests. The results demonstrate the viability of implementing LPWANs for energy-constrained IoT devices.

Beknopte samenvatting

De wijdverspreide toepassing van apparaten die verbonden zijn met het internet der dingen (*Internet of Things – IoT*) heeft geleid tot een groeiende behoefte aan een duurzame werking van IoT nodes met een laag energieverbruik. Een cruciale factor voor energiezuinigheid is de optimalisatie van draadloze communicatiemogelijkheden binnen een IoT-node. Diverse nieuwe netwerktechnologieën, bekend als *low power wide area networks (LPWAN's)*, zijn ontwikkeld om te voorzien in communicatie met lage bandbreedte, groot bereik en laag energieverbruik. Geen enkel netwerk kan echter als de ultieme oplossing voor alle IoT apparaten worden beschouwd. Dit onderzoek beoogt deze LPWAN netwerken te karakteriseren, met de focus op energieverbruik, en de voordelen te onderzoeken van het combineren van meerdere IoT-netwerken.

De eerste fase van het onderzoek omvat het karakteriseren van de energieverbruiksprofielen van individuele *radio access technologies (RAT's)* onder realistische omstandigheden, inclusief aspecten zoals hardwareontwikkeling en netwerkimplementatie. Factoren zoals datatransmissiesnelheid, dekkingsafstand en vereiste pakketlatentie worden onderzocht om hun impact op de batterij autonomie van een node te begrijpen.

Gebaseerd op de bevindingen van de analyse van enkele RAT's, wordt onderzocht welke voordelen en uitdagingen er zijn bij het implementeren van multi-RAT benaderingen voor de communicatie van IoT-nodes, waarbij twee of meer draadloze IoT-technologieën worden gecombineerd. In dit onderzoek worden twee benaderingen overwogen: het integreren van meerdere RAT's in een enkele IoT-node en het opzetten van multi-RAT-netwerken van IoT-nodes.

Door het integreren van complementaire draadloze technologieën binnen een enkele node ontstaat de mogelijkheid om de meest geschikte technologie te selecteren op basis van specifieke omstandigheden en vereisten. Dit biedt aanzienlijke voordelen, zoals energiebesparing, verbeterde dekkingsgraad en betere betrouwbaarheid.

Aanvullende draadloze technologieën kunnen ook ingezet en gedeeld worden in een groter IoT-netwerk, waardoor een gedistribueerde aanpak mogelijk wordt. Hierbij is het slechts nodig enkele nodes in het netwerk uit te rusten met een cloud-verbinding, die fungeren als centrale datasinks. Om dit te realiseren wordt een netwerkprotocol voorgesteld, waarmee gegevens efficiënt naar de centrale datasinks verzonden kunnen worden. Dit protocol garandeert niet alleen een laag stroomverbruik voor alle nodes in het netwerk, maar vergroot tevens het dekkingsgebied van het netwerk via een multi-hop protocol voor alle nodes.

De onderzoeksresultaten worden toegepast op diverse praktijkvoorbeelden, waarbij de focus ligt op IoT-monitoringsscenario's zoals het monitoren van geluidsoverlast in slimme steden en het bewaken van de gezondheid van bomen en het detecteren van bosbranden in slimme bossen. De resultaten tonen de haalbaarheid van het implementeren van LPWAN's voor IoT-apparaten met een beperkte energievoorraad.

List of Abbreviations

3GPP 3rd generation partnership project.

A-GPS assisted global positioning system.
ABP activation-by-personalisation.
ADC analog-to-digital converter.
ADR adaptive data rate.
AES advanced encryption standard.
AM amplitude modulation.
ARQ automatic repeat request.
AT ATtention.
AWGN additive white gaussian noise.

BER bit error rate.
BLE Bluetooth low energy.
BOD brown-out detection.
BPSK binary phase shift keying.
BW bandwidth.

CAD channel activity detection.
CCDF complementary cumulative distribution function.
CDF cumulative distribution function.
cDRX connected mode DRX.
CE coverage enhancement.
CMOS complementary metal-oxide-semiconductor.
CPU central processing unit.
CR coding rate.
CRC cycle redundancy check.
CSS chirp spread spectrum.

DBPSK differential binary phase shift keying.
DMA direct memory access.
DMAC direct memory access controller.
DRX discontinuous reception mode.

ECC elliptic-curve cryptography.
eDRX extended discontinuous reception mode.
EM electromagnetic.
eNB evolved Node B.
ETSI European telecommunications standards institute.

FDMA frequency-division multiple access.
FEC forward error correction.
FFT fast Fourier transform.
FSK frequency shift keying.

GFSK gaussian frequency shift keying.
GNSS global navigation satellite system.
GPIO general-purpose input/output.
GPRS general packet radio services.
GPS global positioning system.
GSM global system for mobile communications.

I^2C inter-integrated circuit.
IO input/output.
IoT Internet of Things.
IP Internet protocol.
ISM industrial, scientific and medical.

LAN local area network.
LDO low-dropout regulator.
LED light emitting diode.
LoRa long range.
LoRaWAN long-range wide-area network.
LPWAN low-power wide-area network.
LQI link quality indicator.
LTC Lithium thionyl chloride.
LTE long term evolution.

MAC medium access control.

MCL maximum coupling loss.
MCU microcontroller unit.
MTC machine-type communication.
multi-RAT multiple radio access technology.

NB-IoT narrowband IoT.
NPRACH narrowband physical random access channel.
NPUSCH narrowband physical uplink shared channel.

OFDMA orthogonal frequency division multiple access.
OTAA over-the-air activation.

P2P point-to-point.
PA power amplifier.
PAN personal area network.
PAPR peak-to-average power ratio.
PCB printed circuit board.
PDR packet delivery ratio.
PER packet error rate.
PHY physical.
PL path loss.
PLL phase-locked loop.
PLR packet loss ratio.
PMU power management unit.
PRS peripheral reflex system.
PSM power saving mode.
PTW paging time window.

QAM quadrature amplitude modulation.
QoS quality of service.
QPSK quadrature phase shift keying.

RA random access.
RAM random-access memory.
RAR random access response.
RAT radio access technology.
RC radio configuration.
RF radio frequency.
RRC radio resource connection.

RSRP reference signals received power.

RSRQ reference signal received quality.

RSS received signal strength.

RSSI received signal strength indicator.

RTC real time clock.

SC-FDMA single-carrier frequency division multiple access.

SF spreading factor.

SHA secure hash algorithm.

SIB system information block.

SINR signal-to-interference-plus-noise ratio.

SNR signal-to-noise ratio.

SoC system on chip.

SPI serial peripheral interface.

TAU tracking area update.

TDMA time-division multiple access.

TDOA time difference of arrival.

ToA time-on-air.

TSCH time slotted channel hopping.

TTFF time to first fix.

TX transmitter.

UART universal asynchronous receiver/transmitter.

UDP user datagram protocol.

UE user equipment.

UNB ultra-narrow band.

VOC voltatile organic compound.

WSN wireless sensor network.

List of Symbols and Notations

$\alpha_{\text{aggregation}}$	Aggregation ratio
$\mathcal{U}$	Uniform distribution between two values
$\mathcal{X}_\sigma$	Shadowing component of path loss model
ρ	Path loss exponent
$\text{BR}_{\text{UL,bit}}$	Bitrate in bit/s for the uplink channel
BW	Bandwidth (BW)
$\text{LQI}_{\mathcal{R}_n}$	Cummulative link quality indicator (LQI) for a multi-hop route
PDR	Packet delivery ratio (PDR)
PDR_l	Packet delivery ratio (PDR) at link l
PL	Path loss
SF	Spreading factor (SF)
SNR	Signal-to-noise ratio
SNR_{max}	Maximum signal-to-noise ratio (SNR) that can be reported by the receiver hardware
SNR_l	Signal-to-noise ratio (SNR) for a single link l
TP	Transmit power
d	Distance
d_0	Reference distance
E_{TX}	Energy spent during transmit
E_B	Energy spent per payload byte
$m_{\text{aggregated}}$	Number of sent messages that included aggregated data
m_{total}	Total number of sent messages
M_{n-1}	Number of received messages during the previous aggregation time
N	Noise power
n_{overhead}	Number of bytes in the overhead of a message

n_{payload}	Number of bytes in the payload of a message
n_{total}	Total number of bytes in a message
$T_{\text{a,downstep}}$	Downstep value of the multi-hop aggregation timer
$T_{\text{a,max}}$	Maximum value of the multi-hop aggregation timer
$T_{\text{a,min}}$	Minimum value of the multi-hop aggregation timer
$T_{\text{a,upstep}}$	Upstep value of the multi-hop aggregation timer
T_{air}	Airtime for a single (long range (LoRa)) packet
T_{measure}	Time between sensor measurements
T_{preamble}	Airtime of the preamble
T_{symb}	Airtime for a single (LoRa) symbol
T_{TX}	Time to transmit a single (Sigfox) message
$t_{\text{exp},n}$	Expiration time of all timers (point in time)
$T_{a,0}$	Initial value of the multi-hop aggregation timer
$T_{a,n-1}$	Previous value of the multi-hop aggregation timer
$T_{a,n}$	Current value of the multi-hop aggregation timer
t_n	Current intended transmission time (point in time)

Contents

Chapter 1

Introduction

The Internet of Things (IoT) has revolutionized the way we interact with our surroundings, going beyond computers and mobile phones. IoT refers to a vast network of interconnected devices, objects, and systems that are embedded with sensors, software, and connectivity, enabling them to collect, analyze, and exchange data. This interconnected ecosystem opens up a multitude of possibilities, allowing for seamless communication and interaction between devices, as well as facilitating intelligent decision-making and automation.

IoT plays a pivotal role in the development of smart cities, where urban infrastructure is integrated with technology to enhance the quality of life for residents. Smart city applications include intelligent transportation systems that optimize traffic flow and parking, waste management systems that monitor garbage levels and schedule collection efficiently, and smart grid systems that optimize energy distribution and consumption. Through the deployment of IoT sensors and connectivity, cities can become more sustainable, and responsive to the needs of their inhabitants.

1.1 Context and motivation

In this PhD, the primary focus is on exploring technologies that leverage IoT for remote sensing applications. From connected rental bikes in smart cities to monitoring agriculture silos and tracking wildfires, IoT is transforming how we

sense the world around us. However, the adoption of IoT technology has also brought forth significant challenges.

The growth of IoT has been marked by significant changes in the number of predicted connected devices. In 2012, IBM predicted that there would be 1 trillion connected devices by 2015, indicating a skyrocketing trend of adoption [Nor16]. However, in 2014, Cisco predicted that only 50 million devices would be connected by 2020, suggesting a significant slowdown in growth [5]. By 2020, the GSMA Association predicted that there would be 25 billion IoT devices by 2025 [21] (not counting mobile phones or computers). As of May 2022, only 14.4 billion IoT devices are connected [24].

According to Everactive, an IoT company specializing in the development of low-power nodes, the limited autonomy of battery-powered devices is the primary restraint to the widespread adoption of IoT devices [14]. For instance, deploying 10,000 devices with a three-year autonomy in an industrial environment would lead to almost 3,333 battery replacements each year (or 9 per day), which is neither economically feasible nor environmentally viable.

One of the most pressing challenges is how to sustain communication between devices located in remote areas with limited access to wireless access points (e.g., Wi-Fi) or power outlets. Such devices rely on batteries for their operation, which makes maximizing their autonomy a critical task. Doing so not only helps mitigate maintenance costs but also reduces the environmental footprint associated with battery disposal and replacement.

To address this challenge, a comprehensive, system-wide approach is required that incorporates hardware, firmware, and communications development. By optimizing each component of an IoT node, it is possible to extend its autonomy, reduce the need for frequent battery replacements, and ensure sustainable, long-term operation.

1.1.1 Use Case Exploration

The applications for remote IoT sensors are diverse and include a range of industries such as agriculture, healthcare, transportation, manufacturing, etc. In this section, the wireless IoT technologies that can greatly benefit these specific use cases are explored. This section focuses on identifying and analyzing the requirements of the use cases considered in this research.

Environmental Monitoring: Forestry, Agriculture, and Smart Cities. Remote sensing technologies via IoT (Figure 1.1) can help foresters and farmers to better understand the surrounding ecosystem, as well as the impact of their practices on the environment.

Healthy forests play a vital role in mitigating climate change. Trees not only act as carbon storage systems, forests also provide a host of other ecosystem services that support human well-being, including air and water purification, biodiversity conservation, and erosion control. In forests, IoT can assist with forest management by detecting and reporting on forest fires [GYI+22; SGA+22], illegal logging [KJD+17; AGZ21; SGA+22], and other threats to forest health [Sha20]. This real-time information can enable rapid response and targeted actions, preventing the spread of fires and reducing the impact of other threats.

Precision farming based on remote sensing via IoT, enable farmers to make data-driven decisions based on actual conditions on the fields [FS20; CLL+19; RAA22]. This helps to optimize the usage of water, fertilizers and agrochemicals, minimizing environmental impact and soil damage, while simultaneously boosting crop yields. By using real-time data to inform their farming practices, farmers can increase efficiency and profitability, while also promoting sustainable and responsible resource management.

In both forestry and agriculture, remote IoT sensors can also help monitor the health of individual plants (crop or tree), alerting managers to signs of stress or disease that may not be visible to the naked eye [GPE+21]. This early detection can lead to faster and more effective treatments, minimizing the risk of widespread damage.

Environmental sensors placed in a smart city landscape are able to collect data on various environmental urban factors, such as air quality and noise levels. By monitoring air quality [DDH+20; JD20], city officials can identify areas where pollution levels are high and take action to reduce emissions, such as diverting traffic away from the city center and create green infrastructure where most needed. Monitoring noise levels [PSL+19; ZLB17] in real-time across a city can identify areas where noise pollution is particularly high. This data can then be used to take action to mitigate the negative impacts of noise pollution on residents.

Transportation: Track and Trace. Through the use of IoT technology, real-time information on the location and movement of vehicles, goods, and people can be easily provided. When it comes to cargo tracking, IoT sensors allow companies to monitor the location and condition of goods in transit. By providing updates on location, temperature, humidity, and vibration levels, companies can

Figure 1.1: Remote connections of sensors in forests, agricultural fields and smart cities via IoT.

take immediate action if necessary, ensuring that goods arrive at their timely destination in the best possible condition [DGG+20; AMH20; TCW+17].

In addition to cargo tracking, IoT technology can also be used to track rental appliances. For instance, in smart cities, IoT can be used to track the location of rental bikes in bike-sharing platforms, making it easy to locate, rent and pay for bikes through a mobile app [LCVD22; RC15]. This not only provides a convenient mode of transportation for city dwellers but also helps to reduce air pollution by promoting the use of bikes instead of cars [Liu18]. These smart bikes can also provide information on the condition of bikes, such as battery levels and maintenance needs, allowing operators to manage their bike-sharing systems more efficiently.

Supply Chain Management. With IoT-enabled inventory management, businesses can optimize inventory levels by automatically triggering replenishment orders, whether trigger level based or prediction based [MBYX22]. This can help suppliers to reduce excess waste and optimize the transportation of goods. For instance, in agriculture, IoT sensors can be installed on silos and other storage tanks to monitor the stock of grain. This real-time data can be used to optimize transportation schedules and ensure that the stock remains topped up, reducing the risk of shortages or overstocking.

Construction: Managing Building Stresses. IoT sensor devices can be strategically mounted on or even embedded in throughout buildings to constantly collect data on various environmental and structural factors [MLR22; AMT+20]. Using this data, construction engineers are able to detect any changes or

abnormalities that may indicate the presence of damage, e.g., the evolution of cracks that could potentially signal unstable constructions.

Conclusion: use case requirements. The various requirements discussed in these examples are summarized in Figure 1.2. These requirements can be categorized into six distinct categories with varying priority ratings: data rate, coverage area, energy efficiency (or node autonomy), cost efficiency, quality of service, and latency performance. The graph depicted in Figure 1.2 shows the relative importance of these requirements across different use cases. Notably, high energy efficiency and extensive coverage area are crucial for all these use cases. The importance of other requirements, such as data rate and latency performance, varies among these IoT applications. Low-power wide-area networks (LPWANs) are wireless IoT technologies that cater to extensive coverage areas and long-lasting node autonomy. They prioritize coverage and energy efficiency by capitalizing on the low data rate demands of these use cases.

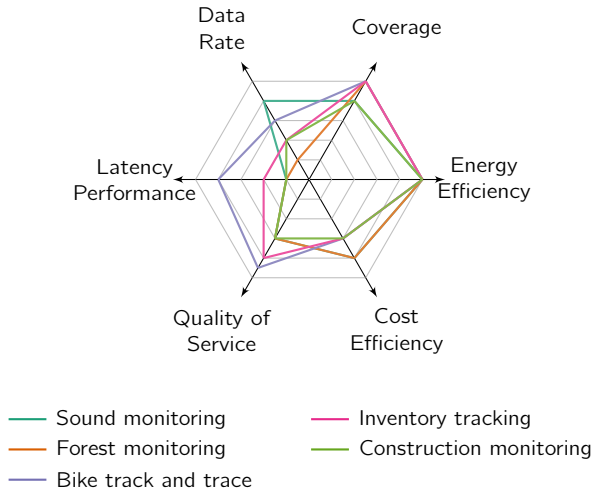


Figure 1.2: Requirements of IoT use cases explored in this PhD.

1.1.2 Internet of Things: PhD Scope

Generally, the Internet of Things (IoT) refers to the network of physical devices, vehicles, home appliances, and other items that are embedded with sensors, software, and connectivity, allowing them to collect and exchange data with

other devices and systems over the Internet, often through the use of a cloud service. This broad term encompasses a wide range of applications that leverage the power of IoT technology. However, in this thesis, the focus is mainly on energy-constrained, uplink oriented IoT nodes. Both hardware and firmware of the IoT device itself are discussed, yet focussing on the communication technology.

In general, when an IoT node is mentioned in this thesis, the following characteristics are implied:

- *Uplink focused messaging:* the communication pattern of IoT devices is characterized by uplink-focused messaging, meaning that communication is only initiated by the IoT node, with downlink communication being rare.
- *Energy-constrained end-devices:* the IoT nodes are limited in battery capacity and have limited opportunities for recharging.
- *Limited periodical messaging:* nodes only measure and potentially transmit sensor data sporadically, resulting in limited messaging frequency.
- *Small payload size:* sensor data transmitted by nodes is characteristically small, in the order of bytes. Combined with the limited messaging frequency, this results in a low data throughput.
- *Low-cost nodes:* IoT nodes are generally deployed in large quantities, which limits the cost of a single node.
- *Long-range connectivity:* nodes are often deployed in remote locations, and need to communicate over long-range communication links of more than 100 m.

1.2 Terminology and Preliminaries

This section provides an introduction of the fundamental ideas, concepts and technologies that form the basis of this book. More in-depth explanations of these concepts are provided in the relevant chapters. Firstly, it introduces some key concepts on how to improve the energy consumption of IoT nodes. Next, it provides an overview of LPWANs. Finally, the section discusses the concept of equipping a node with multiple radio access technologies, which can provide advantages in terms of both wireless connectivity and energy-efficiency.

1.2.1 Energy-Aware IoT Devices

Energy-aware IoT refers to the development of IoT devices and systems that reduce the energy consumption to the bare minimum. Energy consumption is a crucial consideration when designing and maintaining IoT nodes as many devices are battery-powered and need to operate for extended periods (up to 10 years) with limited or entirely without recharging. Multiple strategies exist to improve the battery life of an IoT node.

- *Hardware components:* The hardware components in IoT systems, including sensors, processors, and radios, play a crucial role in energy consumption, and opting for low-power components can mitigate overall energy usage.
- *Power management:* Power management techniques such as turning off the radio or other components of an IoT device for periods when they are not needed helps reduce energy consumption. Also, utilizing sleep modes where the device is put in a low-power state when it is idle reduces idle energy expenditure.
- *Network connectivity:* Using networks that operate on lowest energy levels within the given use-case requirements.
- *Power monitoring:* Energy-aware IoT systems can also benefit from data analysis of their own current draw to optimize energy consumption. By analyzing data from IoT devices, algorithms can identify patterns and make predictions to optimize the energy consumption, particularly for the on-board wireless technology.

1.2.2 Low-Power Wide Area Networks

In this PhD we specifically focus on addressing the challenges faced by energy-constrained remote IoT devices that require connectivity over long distances. Traditional communication technologies such as Wi-Fi or conventional cellular technologies are not feasible options in these scenarios due to their high power consumption. Low-power wide-area network (LPWAN) technology is an ideal solution for these devices, as it combines low-power consumption with long-range connectivity, yet at the cost of a limited payload size, which ranges from tens of bytes to just over one kilobyte. Fortunately, the intended use-cases do not necessarily require higher payload sizes. In environmental use cases, for example, the following payload could be sent: 4 B temperature value, 2 B humidity value, 4 B specific gas sensing values for estimating the considered air quality. Together, this example would only require a payload size of 10 B.

To address energy constrained IoT devices, LPWAN technologies such as long-range wide-area network (LoRaWAN), Sigfox, and narrowband IoT (NB-IoT) have gained significant popularity in recent years. Each of these technologies offers unique features that are well-suited for specific use cases. These are further elaborated upon in this PhD study.

Other LPWAN technologies such as LTE-M have also been developed, but they focus on a market requiring higher throughput and are less energy-efficient [ADA+19]. Additionally, several other LPWAN technologies exist, such as Wi-Fi Halow, DASH7, and Weightless, but they have seen less adoption and are not considered. Overall, this research aims to explore the capabilities and limitations of LPWAN technologies to enable efficient communication for energy-constrained remote sensor devices.

Personal area networks (PANs) or local area networks (LANs), such as Bluetooth 5, Bluetooth low energy (BLE), Wi-Fi, and ZigBee, are not considered LPWANs due to their limited range and/or higher power consumption. These networks are designed to cover limited areas, often centered around a single person or a household, and are optimized for higher data rates. Additionally, at the time of writing they lack the infrastructure required to provide widespread coverage over large geographic areas. In contrast, the evaluated LPWANs options (NB-IoT, LoRaWAN, and Sigfox) already boast the essential deployment infrastructure, whether it's provided by operators or established by the community.

BLE Coded physical (PHY) presents a promising technology, particularly in terms of energy efficiency, positioning itself as a potential competitor to LPWAN technologies. It manages to extend its range compared to standard BLE, yet it remains limited when measured against true LPWAN solutions. Typically, BLE Coded PHY achieves distances of less than 1 km, which, although an improvement, falls short in comparison to LoRa or Sigfox, which can span distances ranging between 3 km to 10 km.

The architecture of IoT networks is generally characterized by a structure as illustrated in Figure 1.3. This architecture comprises three main components: (a) the IoT devices that provide a sensing, measurement, and/or control functionality, (b) a wireless connection to a gateway or base station for information exchange with the remote device, and (c) a server or cloud platform where data can be stored, visualized, further processed and interpreted.

This thesis is primarily dedicated to exploring LoRaWAN, Sigfox, and NB-IoT wireless networks. The following provides a brief introduction to these networks, while a more detailed analysis is presented later in the thesis.

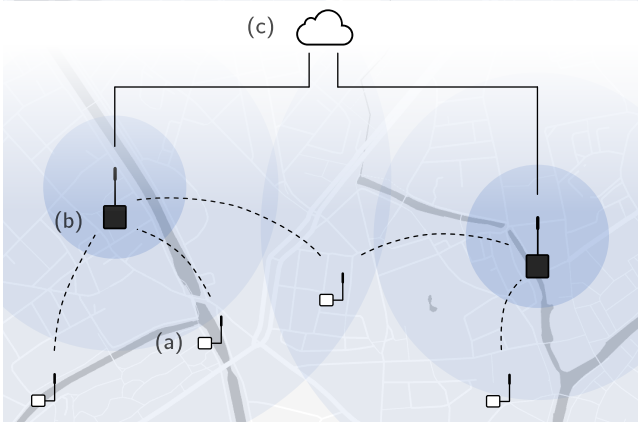


Figure 1.3: The typical architecture connects remote IoT nodes (a) via long-range connectivity to one of more gateways or base stations (b). The interconnected gateways relay the information to the server or cloud resources (c), where data can be visualized and interpreted.

LoRaWAN Long-range wide-area network (LoRaWAN) is particularly suitable in IoT use-cases which require low-power communication while sending small amounts of data. It is a non-cellular network which operates in unlicensed radio frequency (RF) bands. LoRaWAN refers to the full protocol stack, which utilizes the proprietary LoRa modulation technique as PHY layer. LoRa is developed and patented by Cycleo (later acquired by Semtech). The LoRaWAN protocol stack is open-source and maintained by the LoRa Alliance.

Sigfox Sigfox is a proprietary LPWAN that is designed to support IoT use-cases that require sporadic and small message transmissions. Similar to LoRaWAN, Sigfox operates in unlicensed RF bands, and uses a proprietary ultra-narrow band (UNB) communication protocol. This protocol enables devices to send small amounts of data over long distances with minimal energy consumption.

NB-IoT narrowband IoT (NB-IoT) (also known as long term evolution (LTE) CAT NB1) is the first LPWAN network that is targeted to cellular network devices and services. It is developed by the 3rd generation partnership project (3GPP) association and finalized in 3GPP Release 13. The main focus of NB-IoT is to provide cellular, low-power, and low-cost connectivity to devices. NB-IoT also specifically aims to deliver indoor coverage for IoT use-cases.

1.2.3 Multi-RAT for the IoT

Multi-RAT support refers to the capability of an IoT device to utilize multiple wireless communication technologies simultaneously or dynamically switch between them based on prevailing conditions. This approach allows devices to adapt and connect to the most suitable network available, maximizing energy efficiency, connectivity reliability, and coverage.

In this PhD, two approaches to providing multiple radio access technology (multi-RAT) capabilities are considered: providing multiple wireless technologies on a single device, and sharing the connection of a multi-RAT node across a network of IoT nodes.

1.3 Research Questions and Objectives

Despite the growing popularity of the IoT, effective energy management remains a significant challenge, especially in battery-operated devices. There is a lack of in-depth experimental analysis on how to reduce the energy consumption of IoT nodes. This research addresses this gap by examining the following research questions:

Q1 – How can low-power, low-cost and long-range capable IoT devices be designed? (Chapter 2)

Q2 – Can the energy expenditure of IoT LPWAN technologies be characterized? (Chapters 2 and 3)

Q3 – Can the combination of implementing multiple LPWAN technologies on a single IoT device be exploited to support new applications and further increase energy efficiency? (Chapter 4)

Q4 – Can a multi-hop network be leveraged to share multiple LPWAN technologies across an IoT network to increase network coverage and reduce the energy consumption? (Chapter 5)

1.4 Thesis Outline and Contributions

To comprehensively study the design of remote IoT nodes, a three-part investigation into low-power electronics is essential, encompassing low-power

hardware, (firmware) operation, and communication (Q1). A comprehensive behavioral model of a remote, low-power device was derived. Moreover, through experimental measurements using custom hardware components, a fine-grained energy profile was obtained, with regard to the IoT communication technologies in question (Q2). These findings were then integrated into a comprehensive platform, enabling a thorough understanding of energy consumption patterns. Furthermore, to fully exploit the potential of different radio access technologys (RATs) in various environments, a mobile hardware framework was developed. This framework accurately measured and highlighted the impact on energy consumption when combining multiple IoT technologies (Q3). Finally, a comprehensive network protocol was designed to provide shared multi-RAT coverage for nodes within the network, ensuring low-power connectivity, even for remote IoT nodes that can not be reached in a single hop (Q4).

The details of the designed experiments and research methods are extensively discussed in their respective chapters, providing a comprehensive understanding of the methodologies employed. In the following, the conducted work and its notable contributions are succinctly summarized on a chapter-by-chapter basis, providing a concise overview of the key findings achieved in each chapter.

Chapter 2 – Designing Remote IoT Devices. This chapter provides a comprehensive exploration of the background, state-of-the-art technologies, and solutions for remote IoT. A holistic assessment of energy consumption in remote IoT nodes is presented, examining three crucial levels: hardware, (firmware) operation, and communication. The insights gained from this chapter serve as an essential foundation for understanding power-saving techniques in remote IoT networks. These techniques are compiled and tailored specifically to enable efficient energy management throughout the IoT node, including the wireless link.

Chapter 3 – Energy-Based Evaluation of LoRaWAN, Sigfox and NB-IoT. Chapter 3 delves into the experimental assessment of the energy profiles of each LPWAN considered: LoRaWAN, Sigfox, and NB-IoT. A meticulous compilation of these profiles yields an in-depth understanding of their energy characteristics. The findings are then integrated into a framework that enables the calculation of IoT node lifespan for specific use-cases.

The main contributions of Chapter 3 are (i) an experimentally obtained energy profile for LoRaWAN, Sigfox, and NB-IoT, including (ii) an in depth overview on the impact of network parameters and properties on the energy consumption, and

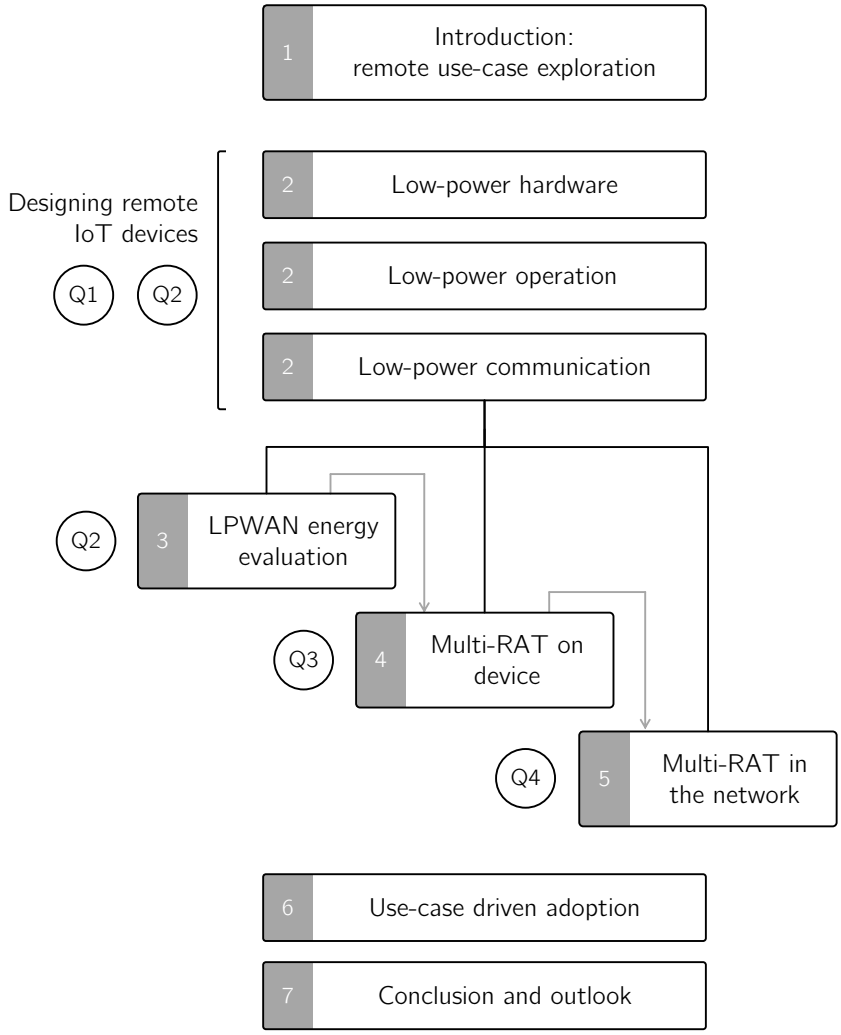


Figure 1.4: Overview of the book's structure, equally acting as a guide on the design process of remote IoT nodes. Note that the chapter titles are shortened for readability purposes.

(iii) a calculator framework to estimate the lifespan of an IoT node, based on the obtained energy profile,

Chapter 4 – Providing Multi-RAT on Device. The study of implementing multiple RATs for remote IoT scenarios is presented in Chapter 4. After the introduction of a custom-made hardware platform tailored to establish an accurate energy profile in real-time, the potential benefits of combining LoRaWAN and NB-IoT are considered.

The main contributions of Chapter 4 are (i) a multi-RAT hardware platform with real-time RAT energy load measurement capabilities, and (ii) an evaluation of a combined LoRaWAN/NB-IoT multi-RAT approach to remote IoT, thereby optimizing, a.o., energy efficiency and support a more diverse set of IoT applications.

Chapter 5 – Multi-RAT supported IoT Networks. When large networks need to be deployed in remote environments, a multi-RAT approach on the network-level may prove to be beneficial. In this concept, each IoT node no longer needs an energy-hungry connection to the cloud, as a single node can share this long link. In this chapter, a network protocol is put in place to create such network: through a LoRaWAN multi-hop protocol, nodes are able to communicate with each other and send data to a central, cloud-connected, node. Special attention is paid to making this network low-power. The presented network protocol is validated based on simulations and an experimental rollout.

The main contribution of Chapter 5 is a novel IoT network that is: (i) based on LoRa multi-hop to reach the central, cloud-connected node, (ii) enables low-power operation by utilizing channel activity detection (CAD) for receiving messages, (iii) improves the energy spent per bit by aggregating messages through multiple hops, (iv) implemented in embedded, hardware independent firmware, and (v) validated based on network simulations, and experimental tests.

Chapter 6 – Use Case Driven Adoption of LPWAN-based IoT. In Chapter 6, an exploration is presented on the selection process for a specific LPWAN approach (whether single-RAT or multi-RAT), through its applicability to various use cases. By studying these use-cases, the valororization opportunities for LPWANs are made clear.

The main contributions of Chapter 6 are: (i) a use-case based presentation on opting for a single-RAT or multi-RAT setup, (ii) a use-case based approach to

selecting one or more suitable LPWAN solutions, (iii) a projection of the gains with respect to energy efficiency when opting for a multi-RAT setup.

Chapter 7 – Conclusion. In the last chapter, a comprehensive review of the conducted work is presented. Additionally, a range of outlooks are provided, highlighting future advances and prospects with respect to both single-RAT and multi-RAT.

Chapter 2

Designing Remote IoT Devices

This chapter delves deeply into the design challenges concerning remote IoT nodes, while Chapter 1 introduces the need for IoT nodes for remote deployment.

First, the main components involved in building IoT nodes are identified, followed by a determination of the primary challenges encountered when designing remote IoT nodes. The process of selecting the appropriate wireless technology is emphasized as a crucial step.

Consequently, a thorough exploration of the state-of-the-art LPWAN technologies, particularly LoRaWAN, Sigfox, and NB-IoT, is conducted. Specific focus is placed on how these technologies are designed for low-power, wide-area applications.

Subsequently, the chapter elaborates on careful component selection, combined with customized software design, to enable low-power operation in IoT nodes. This includes introducing battery-preserving strategies such as "think before you talk" and "race to sleep" as effective measures to prolong a node's autonomy.

This chapter is partly based on the following overview paper:

G. Callebaut[†], G. Leenders[†], J. Van Mulders, G. Ottoy, L. De Strycker, and L. Van der Perre, "The Art of Designing Remote IoT Devices—Technologies and Strategies for a Long Battery Life," Sensors, vol. 21, no. 3, p. 913, Jan. 2021. [CLV+21]

Contributions: The author took the lead in conceptualizing and writing the sections that describe the anatomy of an IoT node and the strategies to extend its battery life with respect to the embedded design. The section on LPWAN technologies was a collaborative effort between the author and the co-authors, with both parties contributing to its conceptualization. The author also made the main contributions to the measurements included in this work.

2.1 Anatomy of an IoT Node

The typical anatomy of an IoT device, and its building blocks, are shown in Figure 2.1. It consists of a controller, a wireless radio module, one or more peripherals (sensors or actuators) and an energy source. In the following paragraphs, some of these components are briefly discussed and the important properties that are needed for IoT devices to operate while using as little energy as possible are highlighted. The strategies to achieve this, are discussed in Section 2.4.

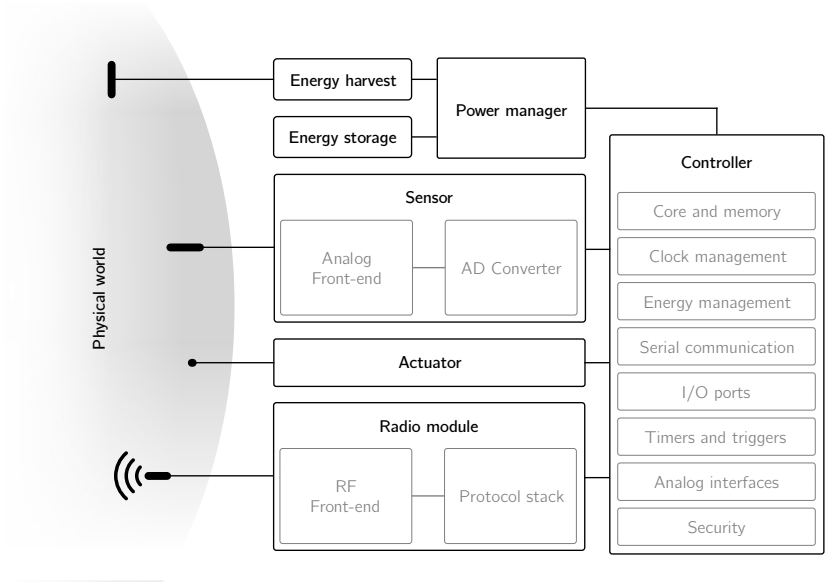


Figure 2.1: Generalized architecture of an IoT node. The complete device is typically battery-powered, optionally assisted by some kind of energy harvesting technology, e.g., a solar panel. Sensors periodically sample the environment for information, which is transmitted wirelessly. A microcontroller has all the amenities to manage the operation and behavior of the node. Actuators, e.g., a LED, allow the node to influence its environment or signal the user.

2.1.1 Controller

The IoT node's operation is controlled by the microcontroller's embedded software, also referred to as firmware. Core clock frequencies can range in order of magnitude from 10 kHz to 100 MHz. The clock frequency affects both processing speed and power consumption (see Section 2.4 for more details). Control over the node's embedded hardware is facilitated by the controller's broad range of built-in peripherals. General-purpose input/output (GPIO) allows for driving basic digital signals, e.g., enable/disable subsystems, as well as reading from them, e.g., status indicators. Hardware support for digital serial protocols [JPC14], e.g., serial peripheral interface (SPI), inter-integrated circuit (I^2C) and RS232 help to interface with subsystems, such as the radio module or the sensors.

Equivalent to the digital communication protocols, analog interfaces are present on a controller to interface with analog sensors. An important example is the built-in analog-to-digital converter (ADC), which in many cases omits the need to include an external ADC on the node's printed circuit board (PCB). To secure the radio communication, dedicated security blocks are often integrated in the controller enabling acceleration of encryption algorithms.[1] Timers, whether or not independently clocked, allow for time-keeping and periodic event firing, e.g., to periodically initiate a sensor measurement. Furthermore, several features specifically aid in energy management of the controller itself. Power for different subsystems can be enabled or disabled and the clock speed can be throttled. The core itself can be put in sleep, while some peripherals remain active. The controller's interrupt system can wake up the core when an event occurs, e.g., a timer fires, or even initiate data transfer between two peripherals, bypassing the core altogether. This is the so-called peripheral reflex system (PRS) system, a network that lets the different peripheral modules communicate directly with each other without involving the core [1].

2.1.2 Sensors

A rich variety of phenomena and signals can be observed using a broad selection of sensors, using different operating principles [Fra16]. Today, the choice is nearly unlimited. Several selection criteria can be applied when choosing a sensor with a certain application in mind.

[1] For example, Silabs' EFM32™ ARM Cortex-based line of microcontrollers integrate hardware acceleration, among others, for AES, SHA and ECC.

Common functional specifications such as supply voltage and required PCB real-estate are highly application dependent. Choosing a sensor that is able to operate on the same supply rail as the microcontroller, generally simplifies the interfacing between the two.

In cases where raw, unprocessed sampling is required, an analog sensor is to be preferred over a digital sensor. Also when ultra-low power operation is required, an analog solution can be advantageous. However, the design of the analog front-end that is responsible for signal conditioning, e.g., amplification or filtering, increases development time, and creates new challenges in terms of power management.

Digital sensors, typically connected to the controller using I^2C or SPI [JPC14], can simplify the hardware design process significantly. Multiple sensors can be connected to the same I^2C-bus, effectively saving pins on the microcontroller. Sensing element, analog front-end and additional logic circuits are integrated on the same device. When the sensor readings are (pre-)processed on-chip before communicating with the microcontroller, it is referred to as a smart sensor [FFL+15]. When selecting a digital sensor, a type with certain integrated features is preferable, as these features help to implement some energy-saving strategies (see Section 2.4). Examples of such features are:

- A sleep or power-down function simplifies power management,
- Automatic sampling (with programmable period) omits the need for extra communication with the microcontroller, i.e., it does not continuously need to "tell" the sensor to start a new measurement,
- Programmable thresholds can further reduce communication between the sensor and the controller. For example, an accelerometer only signals the controller when the measured acceleration is above a certain G-force.

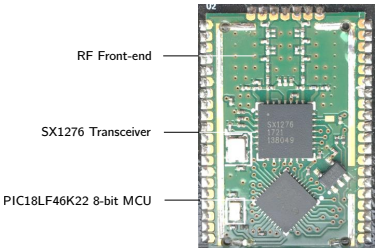
While some applications require only measurements of a single parameter, many, e.g., environmental monitoring, involve the collection of several parameters. It is common to find sensors that integrate a set of related measurements. For example, the BME680 is a digital low-power gas sensor that is able to measure air temperature, humidity and pressure, specifically suited for indoor operation [2]. Furthermore, it is able to detect voltatile organic compounds (VOCs), making it suitable as an (indoor) air quality sensor.

Quite often applications need knowledge of where measurements take place. If sensor nodes are mobile, or if deployment is random, e.g., by an airdrop, the position of the sensor is a measurement in its own right. To locate an IoT node, an obvious course of action is to incorporate a global navigation satellite system

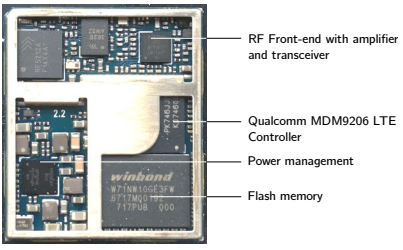
(GNSS) receiver into the device. A GNSS receiver can locate the node by using one of the GNSS satellite systems (e.g., global positioning system (GPS) or Galileo). The module can compute a real-time position using satellite information: latitude, longitude and altitude within a radius of 10 m or better [DR01]. However, localization comes with a non-negligible energy cost, primarily due to the long setup time. More energy-efficient approaches exist by using native geolocalisation based on the uplink messages. The advantage of this approach is that this information comes "free", i.e., it can be extracted from the communication needed to send the sensor measurements.

2.1.3 Radio Module

The radio module is responsible for exchanging information wirelessly. Two options are available, i.e., a transceiver or a modem. The former supports only the PHY layer of the protocol. Often, the RF front-end, including the power amplifier (PA), is integrated on chip. The medium access control (MAC) layer needs to be implemented on the controller. A (wireless) modem, on the other hand, simplifies interfacing by implementing the full protocol stack. A modem is typically controlled by means of ATtention (AT) commands. The benefit of using transceivers, is that only one microcontroller is present on the device, while the modem also incorporates a controller to handle the MAC. Although these embedded controllers are also ultra low-power, it still yields a non-negligible energy penalty. Despite this, modems significantly lower the development effort as no MAC layer protocol has to be implemented on the microcontroller. This also ensures that the node respects the protocol specification. Both approaches can be combined where the microcontroller – in the system on chip (SoC) – can be directly programmed with user-specific code, e.g., Nordic's nRF52832 Bluetooth 5.2 SoC.



(a) Components inside the Microchip RN2483 LoRaWAN modem.



(b) Components inside the Quectel BG96 NB-IoT modem.

Figure 2.2: Modems combine RF front-end, transceiver and MAC layer implementation on a controller in one easy to use module. These examples show the internal components of two popular IoT modules.

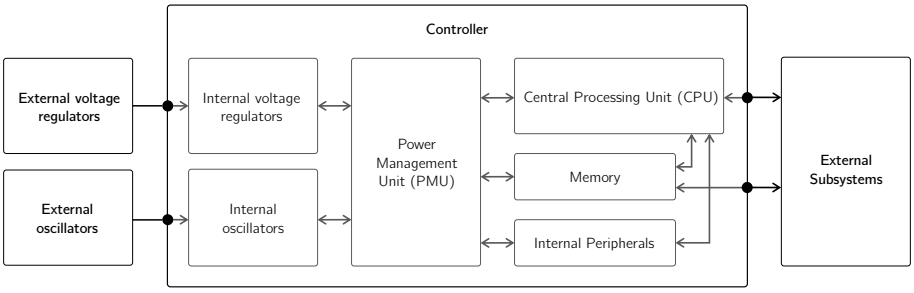


Figure 2.3: The internal power management unit (PMU) of a microcontroller supervises other microcontroller unit (MCU) building blocks. Depending on the current controller state, different voltage regulators are connected or disconnected. The firmware on the CPU can, in turn, enable or disable external subsystems, such as, sensors or the modem.

2.1.4 Power Management

Power management on the IoT node consists of two main parts. A first part is the voltage regulation, which ensures that the voltage input, either from the battery, energy harvesting or any other power source, is converted into the operating voltage for the node's electronics. Most often this includes a low-dropout regulator (LDO) voltage converter or switching mode voltage regulator. Note that these electronics might require different operating voltages. In any case, these voltages are distributed to the controller and all peripherals through the so-called supply rails.

A second part is the power switching, which is typically governed by the firmware running on the central processing unit (CPU) (see Figure 2.3 for the blocks involved). The power switching, in turn, can be divided into two parts. Firstly, the external subsystems, such as sensors, modem or even complete supply rails, can be enabled or disabled by the CPU. This typically involves load switches or control signals that are connected to the controller's GPIO. Of course, such type of control requires that the appropriate measures are taken when designing the hardware.

The power inside the controller is managed through the power management unit (PMU). One task of the PMU is providing a constant supply voltage to components inside the controller. Another task of the PMU is to control power distribution inside the controller, i.e., cut-off or provide power to certain blocks. By doing this, different power states are supported each providing different features. For the well-known Arm-based types of microcontrollers, which are much-used in IoT designs, these are the so-called energy modes.

2.1.5 Energy Provisioning

Typical IoT nodes rely on batteries as their main energy storage and provisioning solution. When selecting a battery, one should first decide on a rechargeable or non-rechargeable battery based on the applicable use-case. Rechargeable batteries are especially suited in combination with energy harvesting techniques when used in remote IoT applications. However, rechargeable batteries generally have a limited autonomy compared to non-rechargeable batteries due to their inherent lower energy density and higher self-discharge rate. Non-rechargeable batteries, while being one time usable only, have some favorable properties compared to rechargeable batteries, not in the least their low self-discharge rate. The generally higher energy density makes them even more suitable for deploy-and-forget IoT applications.

Lithium thionyl chloride (LTC) batteries prove to be a great choice for use as a non-rechargeable battery in remote IoT applications as they have a minimal self-discharge rate, ensuring a long lifespan without the need for frequent maintenance. However, their large internal resistance means they are not well-suited for drawing large power peaks, which are typically required for wireless communication.

To address this issue, a large capacitor or supercapacitor can be connected in parallel to the battery. This allows power peaks to be drawn from the capacitor, rather than from the battery, which helps to preserve the battery's lifespan due to the internal chemistry. Using a capacitor or supercapacitor alongside an LTC

battery [3] is an effective way to optimize wireless communication while maintaining a long-lasting power source.

2.2 Challenges in Connecting Remote IoT Nodes

To maximize the autonomy of long-range IoT devices, the energy budget needs to be managed meticulously, stingy, and correctly: creative design of the nodes is desirable, creative energy accounting is unacceptable. To estimate the expected energy need of IoT devices, the following approach can be taken:

1. Make an inventory of operational modes of the device, including active and idle states,
2. For each of the modes, estimate the power consumption of the node,
3. Calculate the energy consumption by weighing the power consumption with the expected time spent in these modes.

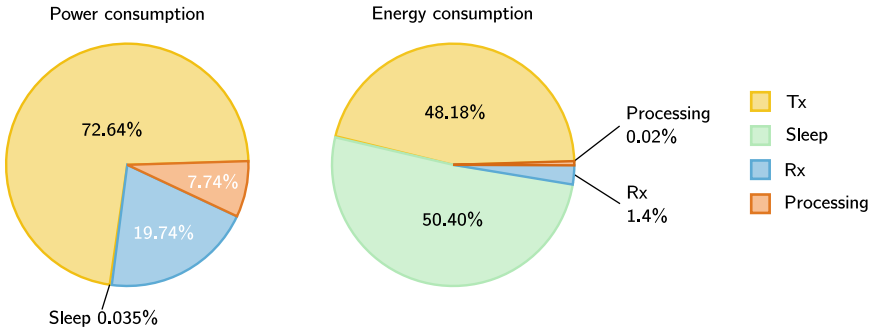


Figure 2.4: Power consumption (left) and energy consumption (right) share of each stage of a LoRaWAN node. The energy corresponds to reading out a sensor and sending a 16 bytes payload packet (SF12) every hour [COv19].

This approach is illustrated for a representative case in Figure 2.4 using LoRaWAN. The results show, in the first place, that the wireless transmitter is one of the big spenders. IoT applications require data to be sent from the device to the server. This implies that the IoT device needs to provide the power to the radio signals to carry the data and spend a relatively large time in this high power mode. This is in contrast to conventional mobile communication networks which are in

general downlink dominated. Figure 2.4 also shows that, for sporadically active nodes, the sleep state has a high impact on the total energy budget, even if power consumption is very low in this mode, as this mode is dominating in time. This highly depends on the considered IoT application and use case, but generally holds true for fire-and-forget applications, where a brief transmission is followed by a prolonged period of sleep.

Selecting the right wireless technology is critical when designing low-power IoT solutions. Hence, the current state-of-the-art LPWAN technologies, in particular LoRaWAN, Sigfox, and NB-IoT, is zoomed in on in Section 2.3. As communication requires a large share of a node's power consumption, battery preserving strategies such as, "think before you talk" and, "race to sleep" are advocated in Section 2.4. These strategies, however, only pay off when combined with a careful hardware design. This design needs to ensure extremely low sleep power, because the consumption in sleep largely determines the node's autonomy.

2.3 Low-Power Wide-Area Networks to the Rescue

In this section we introduce the state of the art wireless communication technologies to connect remote IoT nodes. Herein, we focus on providing insights in specific transmission solutions to overcome the inherent discrepancy between low-power and large coverage, complementary to other overviews [AVT+16; IRS18]. Furthermore, the landscape of current IoT technologies is assessed, considering energy consumption in conjunction with communication characteristics.

The most prominent requirements for serving remote IoT nodes are long-range and low-power connections. LoRa [36] and Sigfox [37] are both LPWAN technologies which gained popularity in recent years and are now key technologies in this domain each bringing specific interesting features. These technologies operate in the unlicensed sub-GHz spectrum due to the favorable propagation characteristics needed for coverage extension. Furthermore, new cellular communication modes and terminal categories are defined, e.g., NB-IoT [19], for machine-type communication and IoT applications. NB-IoT can operate both in sub-GHz and conventional cellular frequency bands.

Long-range and low-power connections require the co-design of both the physical [RDRB18] and medium access control layer [BDWL10]. The PHY layer defines the modulation scheme applied to get bits of information transported by electromagnetic (EM) waves. The MAC layer specifies the manner in which the devices or nodes access the shared medium, i.e., the radio band. It coordinates

the access to mitigate and minimize packet collisions and interference. An optimized design in both layers is required to achieve low-power operation. This section elaborates on how the technologies address the low-power and long-range constraints. A summary of the design choices of each LPWAN technology can be found in Figure 2.5.

2.3.1 Tackling a Node's Low-Energy Budget

LPWAN technologies adopt simple yet adequate modulation schemes and energy-efficient MAC principles to support low-power operation.

2.3.1.1 Low-Energy Physical Layer

The primary objective throughout the evolution of traditional communication technologies has been to enhance throughput. This has been accomplished by introducing increasingly spectral-efficient schemes, thereby packing more information per Hz, evident in the transition from 2G to 3G and 4G, for instance. Information is digitally modulated and can be represented by constellation points on a complex plane, as shown in Figure 2.6 [Skl+01]. The more constellation points, the larger the alphabet of symbols that can be carried by a single sample. The length of this alphabet, i.e., the number of different symbols, is called the modulation order. To increase the spectral efficiency, higher order modulation schemes are used, densifying the constellation points. This makes these schemes more error-prone than lower-order modulation, as additive noise and signal distortions at the receiver cause wrong decisions on which constellation point was actually transmitted. As a result, the utilization of high-order modulation techniques requires a high energy per bit to noise power spectral density ratio (E_b/N_0) to achieve an adequate bit error rate (BER) for high throughput connections.[2] To enhance the BER, one can opt for strategies such as transmitting over shorter links or employing significantly higher power levels. Yet, none of these are feasible nor required for remote IoT devices, which are designed to operate as low-power, long-range devices. To address this specific need, LPWAN technologies deliberately constrain throughput in order to facilitate long-distance communication while maintaining low power consumption. Consequently, they adopt more resilient

[2]This holds true in ideal conditions, assuming the presence of only additive white gaussian noise (AWGN). However, in real-world scenarios, factors such as the typical multipath fading, interference, or shadowing can significantly influence throughput. Achievable throughput is contingent upon a range of factors, most notably the occupied bandwidth and the signal-to-noise ratio (SNR), as defined by the Shannon–Hartley theorem.

Figure 2.5: Overview of approaches of LPWAN technologies to address long-range and low-power requirements. The PHY and MAC is simplified in order to transmit small packets at a low data rate and low transmit power. Note that the frequency and power values referenced in this table correspond to the implementation and regulations in force within the EU as of the time of writing.

		LoRaWAN	Sigfox	NB-IoT
PHY				
	Modulation scheme	CSS	D-BPSK	BPSK/QPSK (SC-FDMA)
	Frequency (EU)	868 MHz	868 MHz	GSM (e.g., 900 MHz) LTE (e.g., 1700 MHz)
	Bandwidth	250 kHz and 125 kHz	100 Hz	200 kHz
	Transmit Power (dBm)	ISM governed: max. 14 (node) / 27 (gateway)		14 / 20 / 23
MAC				
	Protocol overhead			
	Initial Access	None (ABP)/Low (OTA)	None	High
	Uplink Packet	13-28 bytes	14 bytes	IP-based (depends on higher layer protocols)
	Collisions			
	Freq. div.	No	Yes	
	Space div.	Yes	Yes	n.a. (grant-based)
	Time div.	No	Yes	
	Overhearing	No	No	No (grant-based)
	Adaptive PHY control	Yes (ADR)	No	Yes (CE levels)
	Maximum payload size	∼250 bytes	12 bytes	1600 bytes

but less spectrally efficient modulation schemes. The impact of the SNR on the constellation points is depicted in Figure 2.6. We illustrate the sensitivity to AWGN for binary phase shift keying (BPSK), transmitting one bit per symbol with a 2-symbol alphabet, and 64 quadrature amplitude modulation (64QAM), transmitting six bits per symbol with a 64-symbol alphabet [Skl+01]. Here, we adopted typical SNR values [CV20] for long-range (−20 dB at 4 km) and a short-range (6 dB at 20 m) LPWAN connections. LTE (4G) defines a good quality link if the SNR is equal or higher than 20 dB, which is used as a baseline for comparison.

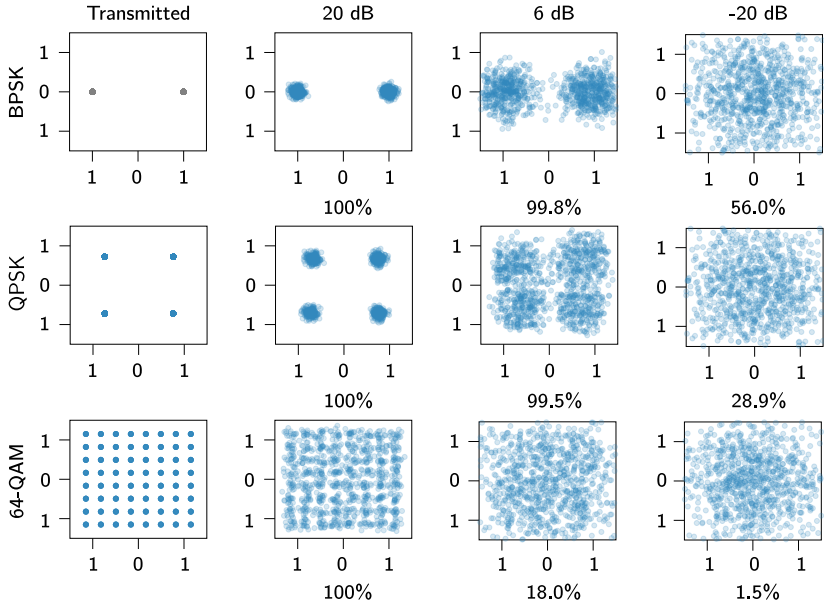
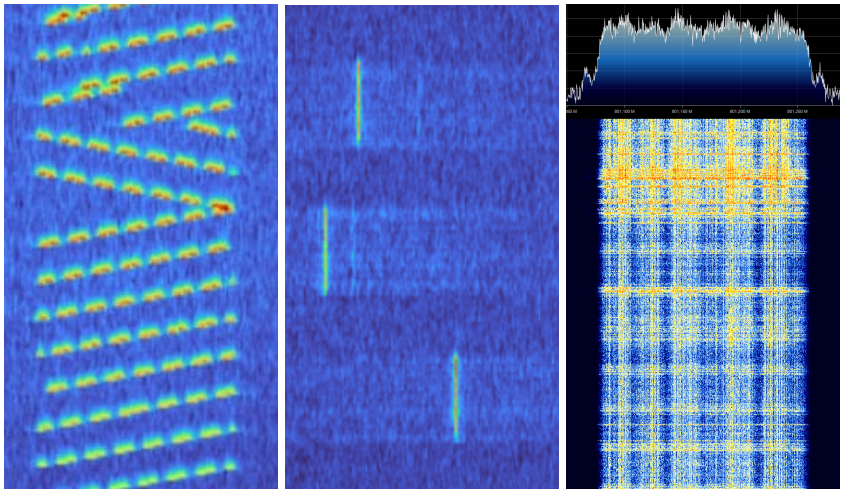


Figure 2.6: Constellation diagrams of BPSK, QPSK and 64-QAM modulated symbols at different SNRs including the percentage of successfully demodulated symbols. Both BPSK and quadrature phase shift keying (QPSK) are used for uplink communication in NB-IoT, depending on network configuration and signal conditions. QPSK and 64-QAM are used in LTE, in addition to 16-quadrature amplitude modulation (QAM), depending on the SNR value. These constellation diagrams demonstrate that low-order modulation schemes are vital in order to work in low-SNR environments or when the transmit power levels are low.

Figure 2.6 demonstrates why low-order modulation schemes are required for low SNR scenarios. Furthermore, using low-order modulation schemes requires less complex hardware, reducing the power consumption of the radio hardware. In particular for transmitting high peak-to-average power ratio (PAPR) signals over a long range, the power amplification stage takes up more than 70% of the power budget, and achieves an efficiency of typically less than 30% because of

the need to operate with significant back-off [CGB05; AGD+11; LLW+11]. A PA amplifies the input signal to a higher power, aiming for linearity to prevent signal distortions. However, practical PAs have a non-linear region near saturation, introducing distortions. To mitigate this, the input signal's power level is reduced, leading to higher required back-off. This reduces efficiency, as PAs operate most efficiently near saturation.

LPWAN technologies mitigate this power inefficiency, by using constant envelope modulation schemes, where the amplitude of a signal is kept constant. To illustrate this, in frequency shift keying (FSK), an alphabet of symbols that have different discrete frequencies of their carrier signal is used [Skl+01]. This scheme does not alter the amplitude of the signal, relaxing the PA constraints. Furthermore, due to the reduced data rate, LPWAN technologies limit the payload size in order to lower the transmit duration. Besides a small payload size, the transmission power is also limited in the unlicensed bands – to comply with regulations.



(a) LoRaWAN (125 kHz) (b) Sigfox (100 Hz) (c) NB-IoT (180 kHz)

Figure 2.7: Illustration of the spectral shape of LoRaWAN, Sigfox and NB-IoT via a measured waterfall spectrum. The vertical axis represents the time and the horizontal axis the frequency domain. Due to the difference in time and frequency allocation, the figures have not the same time and frequency scale.

Implementations in LPWAN.
Each technology adopts different strategies, e.g., wideband vs. narrowband. This is illustrated in Figure 2.7 where a measured waterfall spectrum of a LoRaWAN, Sigfox and NB-IoT message is depicted.

LoRaWAN. In LoRaWAN, LoRa and FSK are used at the PHY layer operating at 433 MHz and 868 MHz in Europe. FSK is used for short-range communication. The proprietary modulation technique LoRa is based on chirp spread spectrum (CSS), in which information is decoded by means of chirps. A single chirp is a sinusoidal signal whose frequency increases (up-chirp) or decreases (down-chirp) over a fixed bandwidth. In LoRa, each data symbol is represented by a cyclic shifted up-chirp over the frequency interval

$$\left(f_0 - \frac{\mathrm{BW}}{2}, f_0 + \frac{\mathrm{BW}}{2}\right), \tag{2.1}$$

where f_0 is the center frequency, and BW the selected bandwidth. The bandwidth is typically selectable between 125 kHz, 250 kHz, and 500 kHz, depending on the region and frequency plan. The spreading factor (SF) determines the number of bits represented by a single symbol and, consequently, determines how much information is spread over time. The spreading factor typically ranges from 7 to 12, as depicted in Table 2.1. The bandwidth BW is divided in 2^{SF} chips, each representing a single symbol. The starting point of each cyclic chirp, which also determines the position of the discontinuity because of the frequency wrap around in the chirp, determines the data symbol that is represented [Van17]. The duration of one symbol (or chirp) is given by:

$$T_{\mathrm{symb}} = \frac{2^{\mathrm{SF}}}{\mathrm{BW}}. \tag{2.2}$$

In other words, by increasing the spreading factor, the energy per bit is increased. This subsequently improves the SNR, yielding a higher sensitivity (see Table 2.2). The receiver sensitivity is a measure of the minimum required received RF power in order to be still able to demodulate the signal. A trade-off between range and energy consumption can be made by changing the spreading factor (see Figure 2.8). Due to the additional gain by spreading the signal and having a constant amplitude envelope, inexpensive low-power high-efficiency PAs can be used.

Sigfox. The Sigfox network uses different modulation schemes and frequency ranges for uplink and downlink communication[3]. For uplink communication Sigfox uses an ultra-narrowband BPSK modulation in combination with pseudo-random frequency hopping. Data is modulated by differential binary phase shift keying (DBPSK) at 100 bps, generating a 100 Hz signal.[4] Opposed to BPSK, where

[3] To simplify hardware requirements of IoT nodes, gaussian frequency shift keying (GFSK) is used for downlink communication.

[4] This is the case for Sigfox Radio Configuration 1, which is mainly used in Europe or the USA.

Table 2.1: Maximum data rates and payload sizes for commonly used configurations of LoRaWAN for coding rate 4/5. Note that the stated industrial, scientific and medical (ISM) rates are subject to the limits imposed in the same ISM bands, which, for commonly used LoRaWAN uplink bands, typically state a 1% duty cycle limit per band.

SF	BW [kHz]	Data rate bit/s]	Max frame payload size [B]	ISM max full message rate [msg/h]	ISM effective data rate [bit/s]
12	125	292	51	12	1.4
11	125	537	51	23	2.6
10	125	976	51	51	5.8
9	125	1757	115	53	13.5
8	135	3125	222	54	26.6
7	125	5468	222	97	47.9
7	250	10937	222	195	96.2

Table 2.2: Sensitivity of a LoRa transmission with regard to various SFs at a fixed bandwidth of 125 kHz.

SF	Receiver sensitivity [dBm]	T_{sym} [ms]
SF 7	−123.0	1.02
SF 8	−126.0	2.05
SF 9	−129.0	4.10
SF 10	−132.0	8.19
SF 11	−134.5	16.38
SF 12	−137.0	32.77

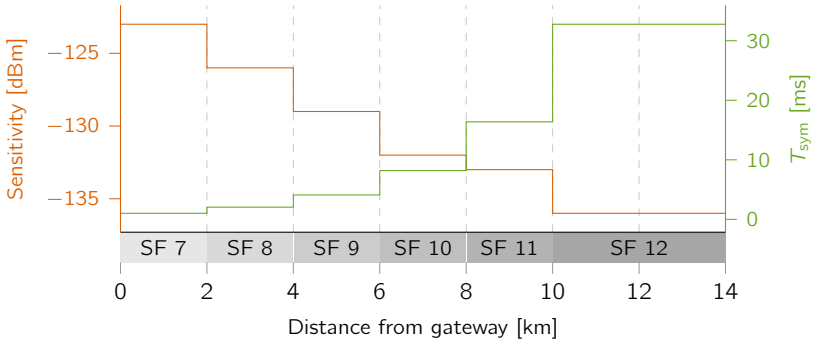


Figure 2.8: The trade-off between LoRaWAN sensitivity (which correlates to range) and the time it takes to transmit a single symbol (which correlates to energy consumption). To estimate the range, a free space path loss model is used to calculate the attenuation between both antennas. The transmit power is set to 14 dBm.

symbols are directly mapped to the constellation points, DBPSK modulates symbols by phase shifts. For example, a binary 1 results in a 180° shift, while 0 does not introduce a phase shift. As a result, the system is more resilient against phase noise, as phase shifts are used rather than fixed constellation points.

NB-IoT. NB-IoT is a slimmed-down variant of LTE that utilizes similar modulation techniques. The devices are multiplexed with single-carrier frequency division multiple access (SC-FDMA). Frequency-division multiple access (FDMA) modulation techniques allow multiple users to transmit data simultaneously over the same frequency band by dividing the available bandwidth into smaller subcarriers and assigning them to different users. NB-IoT uses a narrow band of 200 kHz, which is significantly less than the 20 MHz or more used in traditional LTE networks. In contrast to orthogonal frequency division multiple access (OFDMA), SC-FDMA yields a lower PAPR [BMF+08] and thereby a lower back-off is required, resulting in a more power efficient PA operation.

The uplink in NB-IoT consist of two chanels [CKA+21]:

- *Narrowband physical random access channel (NPRACH)* is responsible for initiating the connection with a cell through the random access (RA) procedure.
- *Narrowband physical uplink shared channel (NPUSCH)* carries both data and control information.

NPRACH always employs single-tone communication, whereas the configuration of NPUSCH depends on the radio conditions and operators' settings, which can be either single-tone or multi-tone. In multi-tone mode, transmission occurs using three, six, or twelve subcarriers to transmit information. On the other hand, in single-tone mode, a single subcarrier is used. The choice between single-tone and multi-tone options depends on the carrier spacing employed by the network. If a carrier spacing of 3.75 kHz is used, only the single-tone option is available. However, if a carrier spacing of 15 kHz is employed, both single-tone and multi-tone options are available [PJJ18].

In single-tone transmission, two modulation schemes are used: $\pi/4$-QPSK or $\pi/2$-BPSK, depending on the network configuration and coverage requirements. In these modulation schemes, the constellation points are rotated every other symbol by the respective phase (hence, $\pi/4$ or $\pi/2$): see Figure 2.9. Each combination of two bits thus occurs twice, this ensures that the phase shift can be limited to a maximum of $3\pi/4$ or $\pi/2$ instead of π with respect to QPSK or BPSK, respectively,

which minimizes amplitude modulation (AM). In multipath spread and fading conditions, these modulations perform better than their classic counterparts, due to an easier detection scheme [CKA13]. When adverse signal conditions are observed, $\pi/2$-BPSK is preferred over $\pi/4$-QPSK due to its better demodulation sensitivity [WLA+17]. On the other hand, in multi-tone transmission, QPSK modulation is always used.

NB-IoT can be deployed in one of three RF band options: in-band operation, guard band operation or stand-alone operation. In-band operation re-uses frequencies which are not used by LTE inside the LTE channel bandwidth. When deployed in guard bands, NB-IoT uses LTE guard band designed as a buffer between strips of spectrum carrying mobile broadband services. In stand-alone operation, NB-IoT is deployed in band, exclusively used for NB-IoT, for example bands that were previously used for global system for mobile communications (GSM) operation [7].

To control the energy drain of the transmission stage in NB-IoT, lower power classes are defined [19], i.e., 23 dBm (Power Class 3), 20 dBm (Power Class 5) and 14 dBm (Power Class 6). Power class 6 was recently introduced in 3GPP Release 14 [HLL+17]. These classes allow to simplify the PA and battery requirements thanks to the lower transmit power. While this brings a clear benefit for IoT devices, currently deployed networks are not fully prepared for these low-power transmissions and therefore these devices can be forced to employ a higher coverage extension, as further discussed in Section 2.3.2.

2.3.1.2 Low-Energy MAC Layer

Collisions, overhearing or idle listening, and overhead are the main causes of energy drainage related to the MAC layer [BDWL10]. Collisions occur when two nodes transmit and the signals overlap in time and frequency whereby the

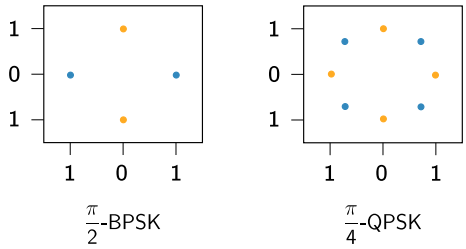


Figure 2.9: Constellation diagrams of $\pi/4$-QPSK and $\pi/2$-BPSK. The constellation points are rotated every other symbol, as depicted by the blue and orange colors.

intended receiver is unable to demodulate the packet. Consequently, the energy consumed by both transmitter and receiver is wasted. Idle listening or overhearing is caused when a device is listening for downlink communication when there is none. Lastly, overhead due to additional signaling or protocol overhead, introduces an extra energy drain. To address these challenges, LPWAN technologies apply simple MAC schemes, as illustrated in Figure 2.10. The communication is mostly device-initiated, meaning that an uplink message is transmitted when the device has some data to send. The remainder of the time the device is kept in sleep mode. Overhearing and idle listening is particularly important in technologies operating in the unlicensed bands where everyone – respecting the regulations – can freely communicate. By specifying the start of a receive window, idle listening is kept to a minimum. Furthermore, no or limited signaling is used to request access to the network and the protocol overhead is constrained to the bare minimum in order to send small-sized packets over the network. The considered applications described in Section 1.1.1 often allow for a packet loss to a certain degree in the sensor network. These protocols often follow a 'fire-and-forget' approach where collisions are of insignificant importance. As a result, the use of acknowledgments and retransmissions is less common and application-dependent. [5]

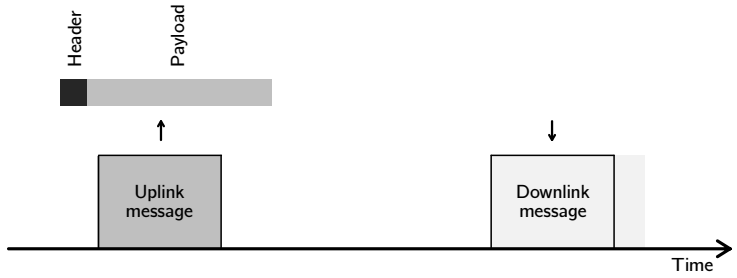


Figure 2.10: Typical MAC scheme for remote IoT devices. The communication is mostly device-induced with a limited payload and minimized protocol overhead. While downlink communication is supported, it is typically restricted to a couple of messages per day.

Implementations in LPWAN.
Both Sigfox and LoRaWAN operate in the unlicensed bands and follow a 'fire-and-forget' approach. They do not employ any specific multiple-access

[5] Not all LPWANs offer an equal amount of quality of service (QoS), e.g., NB-IoT – being a cellular network – does not rely on ALOHA and inherently provides retransmissions and data acknowledgments. Additionally, NB-IoT enables the use of Internet protocol (IP) based communication up to node-level, which can also increase QoS by choosing application layer protocols with a high QoS. In general, QoS can be increased inherently in networks by implementing techniques such as forward error correction (FEC) or automatic repeat request (ARQ).

technique, yet they rely on collision avoidance techniques such as repetitions and narrow-band signals in Sigfox [LPP19], and spread-spectrum in LoRaWAN to mitigate interference. Therefore, LoRaWAN and Sigfox are best effort LPWAN protocols without any guaranteed delivery. Both LoRaWAN and Sigfox incorporate acknowledgments into their protocols to some extent. However, Sigfox provides significantly fewer acknowledgments compared to LoRaWAN. Moreover, both technologies specify the time and frequency of a receive window in order to limit idle listening.

LoRaWAN. The LoRaWAN protocol stack is defined in the LoRaWAN specification [32], and it consists of a MAC and network layer on top of the PHY LoRa. LoRaWAN targets three types of device classes: A, B, and C, each targeting different use-cases. Each class has a vastly different energy profile, with higher classes exhibiting higher energy consumption. The most energy efficient class is class A, which is designed for applications primarily focused on uplink communications, such as sensor readings. Receiving downlink messages is only possible in class A, in a short interval after an uplink message has been sent. For the purposes of this PhD study, which focuses on energy-constrained devices with an emphasis on uplink-centric communication, we limit the energy profile analysis to class A devices within LoRaWAN.

LoRaWAN class A devices (Figure 2.11a) open a first receive window one second[6] after the uplink message, where the downlink message has to use the same spreading factor and frequency of the transmitted packet. In case no preamble is detected, a second window is opened typically two seconds after sending the uplink message at a default frequency and spreading factor. The default frequency (or channel) and spreading factor is 869.525 MHz and SF12 [36].

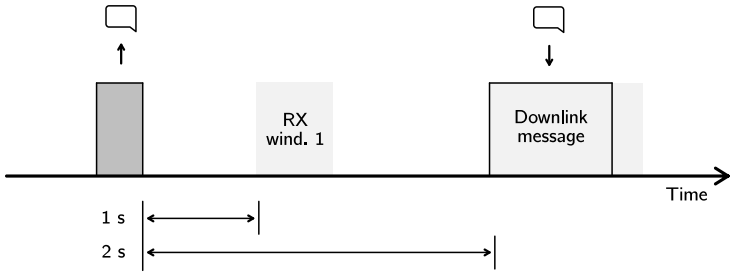
It's worth noting class B devices introduce scheduled downlink opportunities, independent of uplink communication, enabling more real-time interactivity with the devices. Class C devices offer a continuous receive window, making them suitable for applications that require nearly continuous communication.

The length of the uplink payload in LoRaWAN depends on the data rate (i.e., SF) used for transmission (see Table 2.1). At the slowest data rate, the maximum payload size is 51 bytes. The maximum payload size for faster data rates is 222 bytes.

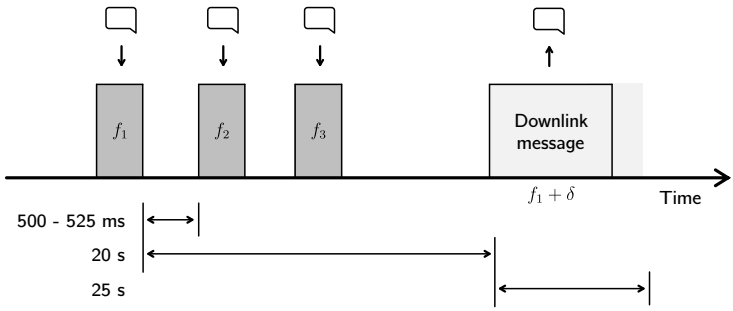
[6]The default delay for opening both receive windows is region and operator specific. The delay can be changed by the network operator through the MAC command *RxTimingSetupReq* The typical value is one second.

Sigfox. Sigfox also supports bidirectional communication, while restricted to a maximum of four downlink messages per day. In contrast to LoRaWAN, Sigfox opens only one receive window (Figure 2.11b). After receiving a downlink message, the device responds with an uplink confirmation message [GVV+19].

The Sigfox message protocol is designed to support small amounts of data with minimal energy consumption. The protocol consists of a binary frame that includes the payload and various control fields. The payload can range from 0 to 12 bytes, depending on the type of device and the application.



(a) LoRaWAN opens two receive windows after a transmit message. The first window utilizes the same data rate as the transmit message. The second window employs more robust settings to increase the chance of reception.

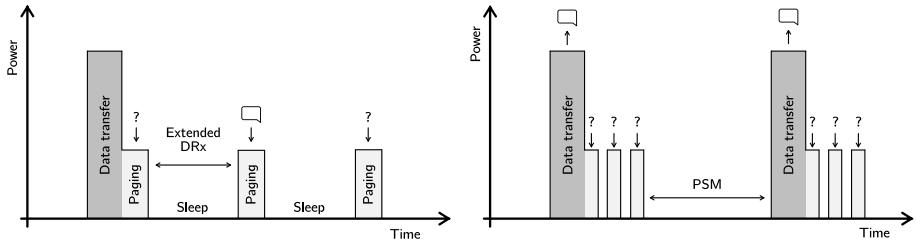


(b) Sigfox transmits three duplicates at different frequencies and different time instances. Limited downlink is possible and is received at the initial transmit frequency plus an offset.

Figure 2.11: Medium access control mechanism for LoRaWAN and Sigfox.

NB-IoT. As NB-IoT is a derivative of LTE, it employs a more complex random access procedure [MPB19]. The node first has to request access prior to transmitting an uplink message. The base station responds with a time and frequency slot where the node gets the opportunity to transmit. Furthermore,

in contrast to LoRaWAN and Sigfox, NB-IoT is fully bidirectional, and not uplink-focused. Both node-terminated (paging) and node-induced traffic are supported, as illustrated in Figure 2.12. As paging and maintaining an active connection can be highly energy consuming, NB-IoT introduces extended discontinuous reception mode (eDRX) and power saving mode (PSM). Both mechanisms are depicted in Figure 2.12. Extended discontinuous reception mode allows a node to listen less frequently to downlink messages. These values are provider-specific, but they have to support at least an eDRX time of 40 min [LCVD20], compared to the maximum of 2.56 s in LTE. A higher energy reduction is achievable by employing PSM. In PSM the node notifies the network that it is going dormant and negotiates the duration of the hibernation. During this period, the node cannot receive any downlink messages but can transmit a packet without having to reconnect to the network. The maximum allowed time is 413 days [LCVD20]. In this manner, applications sending one packet per day can configure a PSM of 1 day. In case a critical message has to be sent, the node can initiate a transmission without having to wait till the end of the agreed-on hibernation period.



(a) Listen more infrequently to downlink message with eDRX.

(b) Hibernate between data transmissions with PSM.

Figure 2.12: Energy saving strategies employed in NB-IoT, i.e., extended discontinuous reception mode and power saving mode, respectively.

Prior to transmitting messages, nodes have to join a network. This initial access overhead is minimized in LPWAN. For example, Sigfox does not use any joining procedure, while LoRaWAN supports both non-initial access and initial access protocols, i.e., activation-by-personalisation (ABP) and over-the-air activation (OTAA), respectively. In NB-IoT, a node has to register to the network after a power-down. Fortunately, through PSM a node does not have to re-join the network after a wake-up. The impact of these configurations is studied in Chapter 3.

2.3.2 Tackling a Node's Long-Range Connection

The communication range of LPWAN technologies is also extended by a co-design of PHY and MAC layer approaches: (i) utilizing low frequency bands, (ii) applying modulation schemes which are less sensitive to a low SNR, (iii) adopting MAC schemes to increase the probability of successfully receiving a packet and (iv) dynamically adapting to the link conditions.

2.3.2.1 Long-Range Physical Layer

In order to extend the range, LPWAN technologies utilize sub-GHz frequencies. The benefits of lowering the carrier frequency is illustrated by the Friis transmission Equation (2.3) [Fri46]. The received and transmit power are denoted by P_r and P_t, respectively.

$$\frac{P_r}{P_t} = G_t G_r \left(\frac{\lambda}{4\pi d}\right)^2 \tag{2.3}$$

The antenna gain is represented by G. The gain of an antenna is angle-dependent. Contrary to common practice, the gain factor in this context does not represent the maximum gain of the antenna. Rather, it specifically indicates the gain in the direction of the other antenna. The antenna gain is the combination of both the directivity of the antenna pattern and the antenna's efficiency, i.e., $G = \epsilon_{\text{antenna}} D$, with $\epsilon_{\text{antenna}}$ the efficiency and D the directivity of the antenna[7]. The distance between the transmitter and receiver is denoted by d. The wavelength of the signal is expressed by λ. The formula demonstrates that the received power (P_r) is proportional to the square of the wavelength (λ). To be complete, the received power depends on the effective area or aperture of the antenna. This area describes the amount of power captured from the EM wave impeding on the antenna and is depending on the wavelength. Therefore, the path loss, defined as the ratio of the received power to the transmitted power, is also dependent on the wavelength. While this formula is defined for ideal free-space conditions, this frequency-dependent path loss also holds in real environments. Moreover, objects appear smaller and have less impact on the signals at lower frequencies because the wireless propagation of the signals is affected by the objects relative to the signal's wavelength.

[7]The directivity of an antenna is defined as the ratio of the radiation intensity at a specific angle (or coordinate) to the total radiation intensity of an isotropic antenna that radiates the same total power P_{tot}, which is equal to $P_{\text{tot}}/4\pi$.

As illustrated in Figure 2.6, low-complexity modulation is better suited for low SNR scenarios. Consequently, coverage is extended by employing low-complexity modulation schemes.

Implementations in LPWAN.
LoRaWAN, Sigfox and NB-IoT typically operate at lower frequencies[8] to benefit from the favorable propagation and adopt low-order modulation schemes. The range can be extended in LoRaWAN by increasing the spreading factor, thereby lowering the demodulation floor. For example, with a spreading factor of 12 a signal with an SNR above $-20\,\text{dB}$ can be demodulated, while for SF7 a minimum SNR of $-7.5\,\text{dB}$ is required in order to demodulate the packet with the same packet error rate (PER). NB-IoT on the other hand switches between modulation techniques, i.e., BPSK and QPSK, depending on the link condition.

2.3.2.2 Long-Range MAC Layer

The discussed LPWAN technologies use three diversity techniques to increase the probability of successfully receiving a packet: (i) time, (ii) spatial and (iii) frequency diversity. In other words, by transmitting on different frequencies, at different time instances and by using multiple receiving gateways, the probability that a packet is lost, is lowered. To further increase the probability of successful receiving a packet, the PHY layer is controlled and adapted to the current link conditions.

Implementations in LPWAN
Both LoRaWAN and NB-IoT employ mechanisms to adapt to the channel conditions. Sigfox on the other hand, utilizes a fixed, predetermined data rate and transmission power for its devices, regardless of the channel conditions.

LoRaWAN In LoRaWAN the adaptive data rate (ADR) algorithm [KHA20b] alters the spreading factor and transmit power to accommodate for more losses in the channel. Furthermore, different spreading factors yield orthogonal packets, e.g., a SF7-modulated packet is not interfered by a SF12-modulated packet. By adopting ADR, the utilized spreading factor in the network is optimized, thereby reducing the number of collisions.

The ADR mechanism in LoRaWAN is controlled by the LoRaWAN network servers. A LoRaWAN message is received by one or more LoRaWAN gateways, which then forward the message to the LoRaWAN network servers. These servers collect

[8] Recently, there has been a growing interest in operating LoRaWAN in the higher 2.4 GHz band to improve latency and throughput as opposed to deployment on the 868 MHz band.

multiple consecutive LoRaWAN messages from the node, including essential metadata such as received signal strength indicator (RSSI). The network servers calculate the median values of relevant metrics and assess both the available link budget and the highest data rate that can be supported. Additionally, a margin for error is considered to accommodate fluctuations in channel characteristics. This information is then used to determine the appropriate data rate for the next uplink transmission from the specific node [KHA20a]. To implement the necessary data rate adjustment, a MAC command is sent down to the end device when the next uplink arrives. This command instructs the node to change its data rate and transmit power accordingly.

When a node is not connected to the network (i.e., no longer receives acknowledgments), it increases the SF and the RF transmit power accordingly after predefined ADR backoff timer, expressed in the number of failed transmitted messages. When communication is (re-)established, the network servers take over via the ADR algorithm.

Note that ADR is primarily suitable for static scenarios due to the inherent slow adaptation of the ADR algorithm to changing channel properties [COP20]. In dynamic environments, relying on ADR can lead to missed messages (when acknowledgments are not enabled) and a possible short-term, higher-than-necessary energy consumption.

An alternative to using ADR is to use blind ADR when no acknowledgments are available in the downlink. In blind ADR, a node profits from the differences in energy consumption of various SFs. The IoT node uses, instead of a single data rate, three data rates and different periodicity depending upon the data rate [FKYP21]. For example, a node transmits the obtained (sensor) data on SF 7, SF 10, and SF 12 with different periodicity: transmitting on SF 7 once every 15 minutes, SF 10 once every 30 minutes, and SF 12 once every hour. Hereby a low energy consumption of the application is retained, while maintaining frequent updates.

Sigfox. Sigfox sends three duplicates of the same packet, as depicted in Figure 2.11. Each packet is sent on another frequency at another time instance. Sigfox operators also try to densify the network so that all packets are received by at least three gateways.

NB-IoT. In NB-IoT three coverage enhancement (CE) levels are defined, i.e., 0, 1, and 2, or normal, robust and extreme respectively. The impact of the CE level

is summarized in Table 2.3. In the highest CE level (CE level 2) 128 repetitions, a higher power class (23 dBm) and single tone transmissions are utilized to extend the coverage. The maximal tolerable loss or maximum coupling loss (MCL) is defined for each of these coverage levels, i.e., 144 dB, 154 dB and 164 dB. By implementing these coverage classes, the MCL is improved by 20 dB compared to traditional general packet radio services (GPRS) or LTE systems.

Table 2.3: NB-IoT parameters, depending on the employed CE level (CEL) [And+19], used in the NPUSCH channel. Note that these parameters are observed by Andrés-Maldonado *et al.* and provide only an example of a possible deployment of NB-IoT.

Parameter	CEL 0	CEL 1	CEL 2
MCL (dB)	144	154	164
Subcarrier spacing (kHz)	15	15	3.75
Modulation	QPSK	QPSK	$\pi/2$-BPSK
Number of subcarriers	12	3	1
Occupied channel bandwidth (kHz)	180	45	3.75
Repetitions	2	16	128

2.4 Strategies for a Long Battery Life

In this section, the exploitation of feature-sets and component selection in combination with tailored software design for low-power operation is elaborated. While in Section 2.3 the energy impact of transmission is considered, the energy consumed in sleep is equally important as evidenced in Figure 2.4 and is studied in this section. Furthermore, an IoT end-device needs to meet the specific requirements of the considered application, as pointed out by the diverse use cases introduced in Chapter 1. This obviously has a major impact on both the hardware and software design. Not only will choosing the right components help a long way towards a small form factor and reduced energy consumption, it will also allow applying certain energy saving strategies to optimize the autonomy out of a single battery charge.

A typical power profile of a sensing IoT node is depicted in Figure 2.13. This periodic profile generally consists of four states: (i) sleep, (ii) wake-up, (iii) processing and (iv) communication. A node should spend most of its time in the sleep mode, which is the lowest possible power state. Several monitoring systems can remain active in this state, but no processing or wireless communications is possible. When an event wakes the node, the controller core and peripherals are woken, followed by a transition period where components, such as analog circuits and oscillators, have to become stable before use. When this period is

completed, the node can enter the 'active' state. In this state, the necessary data is gathered and prepared for wireless transmission. After all wireless communication is completed, the node might do some additional processing, e.g., reconfigure a sensor's settings based on the received downlink message, and goes back into sleep mode. Note that the wireless activity is optional and is not always required – or even desirable – in each cycle, as discussed in Section 2.4.3.2.

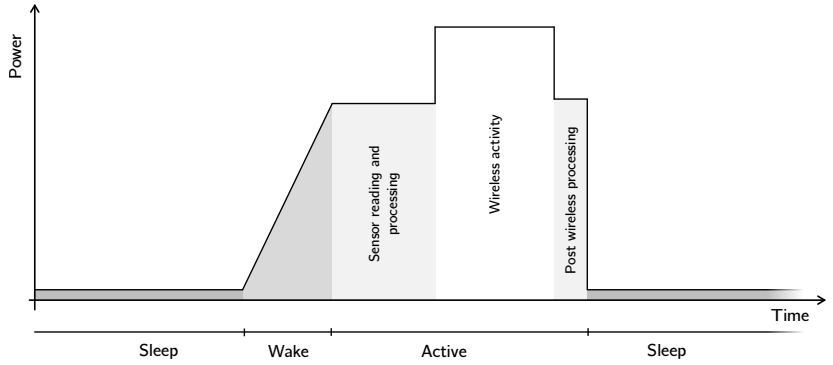


Figure 2.13: Typical power profile of an IoT node. This periodically recurring profile generally consists of four states, i.e., sleep, wake-up, processing and connectivity. A node typically spends most of its time in the sleep mode. The area under the curve yields the energy consumed in each state.

The total energy consumed in one cycle can be noted as the sum of the energy used in each state.

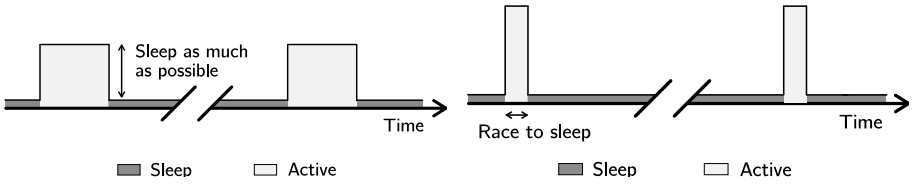
$$E_{\text{total}} = E_{\text{sleep}} + E_{\text{wake-up}} + E_{\text{active}} \tag{2.4}$$

It is clear that to save energy, the power consumed in each of these four main states should be minimized. More importantly, high power states (such as wake or active) should be strictly limited in time. This allows for more time in low-power states (sleep) and, hence, leads to energy savings. In the following sections, we discuss possible energy saving mechanisms in each node state, i.e., sleep, wake-up and active.

2.4.1 Sleep

We suggest to follow the rules "sleep as much as possible" and "race to sleep". Through the former approach (Figure 2.14a) other high-power states should be avoided as much as possible. Despite the possibility of a higher power consumption,

the latter (Figure 2.14b) proposes to consider speeding-up the microcontroller to more quickly enter the sleep state. Moreover, hardware acceleration can further decrease the processing time and hence, increase the time spent in sleep.



(a) *Sleep as much as possible.* Avoid powering unnecessary hardware and fall-back to a sleep state as often as possible.

(b) *Race to sleep.* Utilize dedicated hardware features to reduce time spend in high-power states.

Figure 2.14: Sleep as much as possible and race to sleep strategies.

By limiting the time in active mode to the absolute minimum, the device spends the majority of its time in sleep mode. Therefore, the energy spent in sleep mode is often a dominant parameter in the overall energy consumption (Figure 2.4). To save energy, all peripherals and unnecessary components are powered down. To trigger the controller to wake-up, several mechanisms can be used, i.e., external interrupts, real time clock (RTC) or a watchdog timer. In this part, the impact of the controller, sensors and the radio module on the sleep energy is studied. Based on this, we propose strategies to reduce the energy expenditure in sleep.

2.4.1.1 Controller

The power consumption in modern microcontrollers – and any digital complementary metal-oxide-semiconductor (CMOS) logic – is determined by the switching power (P_{switch}) and static power consumption (P_{static}), as illustrated in Equation (2.5) [HSN04]. Switching or active power refers to the power consumption due to the switching of the digital logic. It depends on the supply voltage U_{DD}, the CMOS switching activity factor α, the load capacitance C_L and the frequency f. In other words, the dynamic power depends on the number of transistors, changing state per cycle, and the core clock speed. Consequently, if more peripherals are being clocked or a higher clock speed is used, power consumption also increases. In addition, the supply voltage has a high impact on both the static and the active power consumption. Hence, the supply voltage

is sometimes lowered to further decrease the power consumption as discussed later-on.

$$\begin{aligned} P_{\text{avg}} &= P_{\text{switch}} + P_{\text{static}} \\ &= \alpha C_L U_{\text{DD}}^2 f + I_{\text{leak}} U_{\text{DD}} \end{aligned} \tag{2.5}$$

However, in sleep mode, almost all clock sources are disabled and thus there is almost no logic active. Only static power remains in sleep mode [4]. Static power refers to CMOS leakage currents. Leakage current is the most dominant factor during sleep and is primarily caused by the output stage, i.e., input/output (IO) pins. For example, a 30-pin input device which has a leakage current specification of 100 nA, can easily consume up to 3 µA while in sleep mode [34]. To minimize leakage currents, controllers incorporate an advanced PMU enabling ultra-low power sleep currents. The PMU cuts the supply to any unnecessary logic modules during sleep. Peripherals such as flash, random-access memory (RAM) and analog interfaces are turned off. Some logic modules can remain active, providing limited functionality during sleep. These remaining units should be clocked from an external, extremely low-power 32.768 kHz clock (commonly found in watches).[9] To further minimize P_{static} in transistors, the MCU operating voltage can be reduced. Some controllers include an on-chip voltage regulator, designed to lower the MCU voltage during sleep to 1.2 V.

Leakage on the IO pins of the controller, can be reduced by configuring unused pins to drive to a high or low state. Digital pins consume the least amount of power when the input voltage is near one of the used voltage rails. If the input voltage on digital ports is near the midpoint between U_{SS} or U_{DD}, the transistors inside the IO port are biased in the linear region,[10] consuming a significant amount of energy. Used IO pins are best configured as high impedance inputs when going to a low power sleep state [15]. These configurations need to be set manually by the developer.

As a consequence of "sleeping as much as possible", the MCU needs to preserve RAM and register values, so it can continue after a sleep period. A microcontroller has special low-power RAM retention schemes to minimize leakage. This is typically achieved either by the use of a very low current latch biasing scheme or by the use of special retention latches that can hold the state without significant leakage [34].

[9] A 32.768 kHz crystal is preferred, as it is low-cost, commonly available and easy to use. The frequency of 32 768 Hz is commonly used because it is equal to 2^{15}. By using a 15 stage binary counter, a precise 1 s period can be achieved.

[10] The linear region of a transistor lies between cutoff and saturation. When a transistor is biased in the linear region, it effectively acts as a voltage controlled resistor.

Most vendors offer different standby or sleep options: variations on RAM retention and brown-out detection (BOD) state [Fer10]. Modern low-power controllers also offer a varying amount of low-power integrated functions that remain active in certain sleep states, e.g., a RTC and low-power serial communication. Enabling these functions increases the power consumption during sleep. Some examples are included in Table 2.4. It is obvious that unused and unnecessary device functions, are best left disabled to minimize power consumption.

Table 2.4: Overview of typical current drawn by optional peripherals in sleep mode. MCU supply voltage is 3.3 V.

Parameter	ARM M0+ (ATSAMD21) [25]	8-bit PIC (PIC16F1717) [31]	16-bit PIC (PIC24FJ128GA310) [30]
BOD (µA)	0.132	0.8	0.07
WDT (µA)	0.007	0.5	0.8
32kHz RTC (µA)	0.056	1.3	0.4

2.4.1.2 Power Management

The design needs to incorporate mechanisms to switch off peripherals in order not to waste idle energy. This is certainly the case when entering sleep mode. Power provisioning and controlling is done by the controller and the on-board PMU (Figure 2.3).

Low-current peripherals (<20 mA) can be directly powered from IO pins. The microcontroller can then completely power off the peripheral during sleep, thereby entirely eliminating any sensor sleep currents. For high-power hardware, such as radio modules, IO ports are not able to support the current drawn in active mode. Therefore, external digital load switches can be incorporated in the design of the IoT node. The leakage current of these switches are important to minimize the energy drain in sleep mode. Another approach, when using multiple voltage rails for multiple peripherals, is to disable any unused voltage rail. By using an LDO with an enable pin, one could instantly switch off any peripherals, connected to that voltage rail. All microcontroller-based power management has to be implemented by the developer through a careful hardware and firmware design.

When it comes to power management within the controller, an essential component employed is the PMU. The PMU effectively handles power distribution, ensuring a consistent supply voltage to all internal components of the controller. Additionally, the PMU controls the power distribution inside the controller, i.e., cut-off or

Table 2.5: Energy modes provided by Silabs' EFM32 ARM-based line of microcontrollers [35]. The listed current consumption is a minimum and increases further with each enabled peripheral. Because high frequency clocks are enabled (and required) in EM0 and EM1, the current draw is a function of the clock frequency.

Energy Mode	Current consumption	Capabilities (non-exhaustive list)
Energy Mode 0 Active/Run Mode	180 µA/MHz	Full capabilities. High performance CPU and with all peripherals available (if enabled).
Energy Mode 1 Sleep Mode	45 µA/MHz	CPU disabled, all peripherals available (if enabled). PRS combined with DMA enables data transmission between peripherals without CPU intervention.
Energy Mode 2 Deep Sleep Mode	0.9 µA	No high frequency oscillators, which means not timers or continuous sampling of the ADC. RTC on a low-frequency oscillator. Low-Energy UART, I^2C (slave operation), Analog Comparator, GPIO.
Energy Mode 3 Stop Mode	0.6 µA	Full RAM retention, no RTC Watchdog timer, I^2C slave operation, GPIO, Analog Comparator.
Energy Mode 4 Shutoff Mode	20 nA	No RAM retention, only asynchronous wake-ups possible from GPIO or reset.

provide power to certain blocks. By doing this, different power states are supported each providing different features. For the well-known ARM-based types of microcontrollers, that are much-used in IoT designs, these are the so-called *energy modes*.

Table 2.5 provides an overview of the energy modes provided by Silabs' EFM32 ARM-based line of micrcontrollers.The goal of these energy modes is to offer an easy way for programmers to trade between energy-efficiency and functionality. A "lower" energy mode offers more functionality, but at the cost of more energy being spent.

- When the CPU is executing code, the controller is in energy mode 0. All peripherals can be enabled and the clock speed can be throttled to speed up processing. Enabling extra peripherals, of course, further increases the current draw,
- In energy mode 1, the CPU clock is disabled. However, all peripherals are still available and they can be configured to work without intervention of the CPU. For example, a timer can be used to periodically start sampling of the ADC which, in turn, uses DMA to store its samples immediately into memory for later processing by the CPU,

- This mechanism can not be used in energy mode 2, as no high-frequency clocks are available. Only low-power peripherals are available and periodic wake-ups of the CPU can be triggered by using a RTC, running on a low-frequency oscillator,
- In energy mode 3, this RTC is disabled. This means that the CPU can only be woken from an external source, e.g., a pin interrupt on the GPIO. However, data is still being retained in RAM.
- Energy mode 4 is the highest energy efficiency mode, offering almost no functionality. The system can only be woken up by GPIO interrupt.

Whenever an interrupt occurs, e.g., a timer fires, the controller is set up back to EM0. This allows the CPU to process the interrupt, and eventually, go back to a higher energy mode. To ensure this way of working is energy-efficient and practical, switching between energy modes takes only a few microseconds.

2.4.1.3 Sensors

When the use case only demands periodic sensor measurements, a sensor can be put in sleep mode for the entire duration of the sleep period. Interrupt-based sensors should be employed to enlarge the sleep period, and hence, no or limited busy waiting needs to be performed. More details regarding the wake-up capabilities of sensors is discussed in Section 2.4.2.2. Based on the application requirements, the sampling frequency should be minimized to extend the sleep period of the sensing device. This again is a responsibility of the developer.

2.4.1.4 Radio Module

Depending on the IoT connectivity technology, radio modules include low-power or sleep states between transmissions. Besides this, several measures can be taken to further minimize the power consumption of the radio module during sleep. It is important to select radio modules with a low-power consumption in all states, but especially in the sleep state, as presented in [MMBB19].

When using LoRaWAN or Sigfox, modems can generally be turned off completely, e.g., by using a load switch in series with the radio module. When using OTAA in LoRaWAN, it is important to store the parameters obtained during the join mechanism in non-volatile memory. In this manner, a potential join request each time the IoT nodes wakes up, is eliminated. As illustrated in [TCLW19], nodes can go out of spec by forcing the node to enter sleep mode prior to opening the

receive windows. Note that this approach deviates from the intended protocol specification, yet can yield significant energy reduction.

When using NB-IoT, the radio module should be put in power saving mode as long and as frequently as possible, as this is the deepest sleep mode possible. In PSM, the connection parameters are maintained so the previous connection can be quickly re-established. These connection parameters are network dependent and cannot be saved in non-volatile memory, as periodic updates are required (tracking area update (TAU)). It is therefore not advisable to completely turn off the NB-IoT modem, as any interruption would result in having to reconnect to the network (thus wasting energy). Notably, this is application-dependent and it is up to the developer to determine the most optimal approach. This trade-off between shutting-down and hibernating through PSM is further studied in Chapter 3.

2.4.2 Wake-Up

After a sleep period, the controller wakes-up to start a measurement and optionally transmits a message. During the wake-up, sensors warm-up, crystals have to stabilize, and radio modules optionally reconnect to network. The trigger, instantiating the wake-up, could originate from a clock source, external event, comparator event or even from an RF signal through the use of wake-up radios [MAG+17; PMK+17]. Depending on the previous state, i.e., sleep or power-down, the wake-up time and resulting energy consumption are different.

2.4.2.1 Controller

When waking from sleep, the microcontroller effectively re-enables its high-speed system clock. The IoT node should support waking-up from sleep mode from either an external trigger (sensor interrupt) or an internal timer. The most flexible periodic wake-up source is an integrated RTC. This RTC enables the core of the microcontroller to be put to sleep for a certain period of time, after which it continues the operation. By using an external trigger source, the controller can go into a deeper energy saving state by also deactivating RTC timers. This lowers the energy expenditure of the microcontroller but requires that the external sources are powered. A common example is the use of an accelerometer, where an interrupt is fired on an input pin of the controller when a certain event is detected. As stated, this allows the controller to enter a deep sleep state, but the sensor still needs to remains active.

2.4.2.2 Sensors

When waking sensors from sleep, it is important to allow enough time for analog circuitry to stabilize prior to measuring. Some electronic components like voltage regulators or voltage references can have a major impact on the settling time. They require significant external decoupling capacitors, and can take milliseconds to settle. Most often, vendors only quote wake-up times for the digital circuitry, while ignoring the time it takes for the analog circuitry to settle. Some sensors inherently need a longer time to wake, e.g., volatile gas sensors and GPS sensors. In [CLV+21], we show that the sensing element of the gas sensor can take up to 92 s to heat up, causing a high energy consumption per measurement.

Volatile gas sensors (like the Bosch BME680) require the sensing element to be heated up to 320 °C before being able to correctly perform measurements. These heaters can draw a significant amount of current, up to 13 mA at 1.8 V [2]. Heating the sensing element to the appropriate temperature can take up to 92 s. The combination of larger heater currents and slow response times results in a high energy consumption per gas measurement (Figure 2.15).

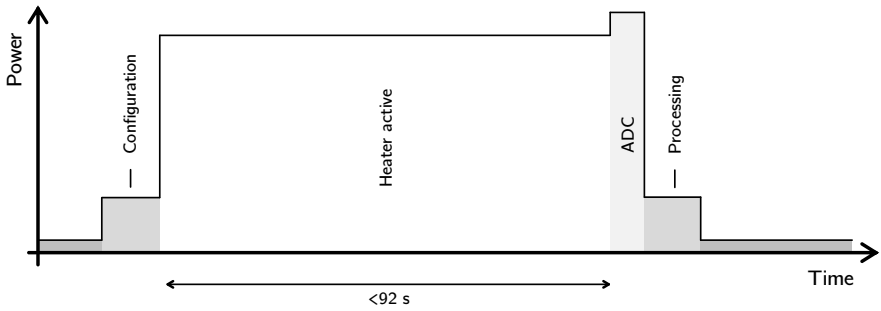


Figure 2.15: Power graph depicting the workings of a 'forced' gas measurement with a BME680 sensor [2]. A sensor remains in sleep and has no active operations until the controller commands, or forces, a measurement. A heated bed is required for an accurate gas value measurement. Heating the sensor element can take up to 92 s, consuming large amounts of energy.

GNSS receivers are also sensors that can potentially require large amounts of energy. They typically can take over 30 s to perform a 'cold' accurate position measurement [LHI08], while drawing considerable currents. A cold GPS measurement starts without any additional location information: no satellite information and no course or previous location. The U-Blox ZOE-M8B (marketed as super low power), draws 34.5 mA (at 1.8 V) while obtaining an accurate GPS fix. To illustrate, one accurate GPS location measurement requires 1.86 J, while

one LoRaWAN transmission, in the worst-case setting (SF12/51bytes), consumes only 1.17 J.

Several techniques, however, exist to reduce the energy consumption of a GPS sensor. Assisted global positioning system (A-GPS) attempts to shorten the satellite acquisition time reducing the wake-up time. A course satellite position is requested in the cloud and loaded into the sensor, thereby reducing the satellite search space and reducing the time to first fix (TTFF) [DR01]. Despite these energy improvements, some applications do not require a very accurate localization. An estimate of the area or region in which a sensor is deployed, is sometimes acceptable. Some LPWAN technologies feature such a coarse-grained localization service. These techniques employ the received data packets to estimate the location of the node. No further signaling is required. Sigfox and LoRaWAN are able to support localization services [HDMH18]. For example, Sallouha *et al.* [SCP17] demonstrate how the LPWAN communication itself can serve as a means of determining whether or not nodes are deployed in each others vicinity. These native geolocalization techniques do not require any wake-up time as was present with GNSS systems.

2.4.3 Active

During the active state the required peripherals and controller are powered-up and active to start or continue the operation. In this phase, one or more sensors are measured and processed. As illustrated by Figure 2.4, in conventional connected sensor systems energy expenditure of processing is much lower than transmitting a packet. As a result, we propose the approach of "think before you talk" (Figure 2.16). It involves on-node processing to verify the meaningfulness of the measured data prior to transmission. It also means that any compressing is advised in order to minimize the payload size of the uplink packet. Accumulating non-time critical sensor measurements could even further lower the energy impact of the transmit phase.

2.4.3.1 Controller

Optimizing energy consumption in the MCU of an IoT device, is often directly related with the "race to sleep" strategy. This can be achieved by using a high clock frequency and dedicated hardware to perform operations more efficiently. A controller can then sooner enter the sleep state, reducing the amount of time in the active state. The power consumed by the controller is directly proportional

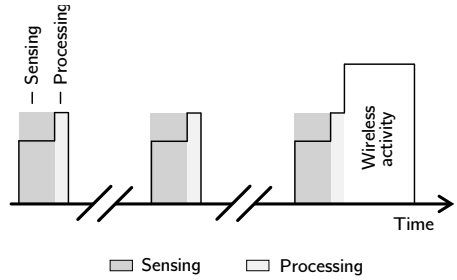


Figure 2.16: *Think before talk.* Check the validity of the sensor measurement and optionally accumulate data prior to transmission. Some sensors do not require the MCU to be active yielding a lower power consumption as depicted by the border.

to the switching frequency, as shown in Equation (2.5). Manufacturers therefore normalize the quoted current draw to a current per frequency (commonly μA/Hz) basis for a specific supply voltage. Most often, the frequency specified represents the system clock frequency. The real performance metric however is the instruction speed: the amount of system clock pulses it takes for certain low level operations to complete. The system clock can run at twice (or more) the speed of the actual instructions, doubling (or more) the effective power consumption.

Most microcontrollers integrate an internal clock divider and/or phase-locked loop (PLL). This lets the software engineer, respectively, slow down or speed-up the system clock. When no high speed operation is needed, a lower system clock can be selected. When high computational power is required, an integrated PLL could be used to boost the clock speed, thereby reducing the time needed for an instruction.

In some controllers, several standard operations can be offloaded to additional integrated hardware components. Security and cycle redundancy check (CRC) calculations are prime examples. By implementing these features in hardware, a lot of computational burden is offloaded from the CPU to a dedicated hardware block. This effectively reduces the amount of CPU cycles it takes to perform these calculations. The CPU needs to wait for the calculation to be completed in hardware. This time is dependent on the calculation itself, but is shorter than any implementation using the CPU [Kam11].

While waiting for results of any peripheral, it is of high importance that the controller can go to a lower energy state, i.e., “sleep as much as possible”. Keeping the CPU active while waiting for results needs to be avoided. The response time of most sensors is typically lower than the instruction speed of a controller. It is clear that the longer the main CPU can spend in sleep, the more energy

savings can be obtained. Three energy effective strategies can be implemented for retrieving the sensor data: polling data transfer, interrupt driven data transfer and DMA [Kam11]:

- Polling data transfer. Data is read using processor instructions. The controller actively polls the sensor for available date. When no data is available, the controller waits for a pre-programmed amount of time. This time is better spent in a low-power sleep state. After this time, the controller checks again whether (sensor) data is ready. When data is available, it is transferred to memory or directly processed by the CPU,
- Interrupt driven data transfer. Data is read using processor instructions, but the controller does not need to check whether data is ready. When data is available, the sensor interrupts any controller state by means of a changing GPIO pin. The controller wakes from sleep or interrupt its current process and acknowledge by starting the data transfer using the CPU,
- Direct memory access. Data is read from the peripheral, not by using processor instructions but by using the integrated direct memory access controller (DMAC). This transfers the incoming data directly to a programmed memory block. The processor continues to wait in a low power until the DMAC calls an interrupt after the last byte is transferred. As IO operations are much slower than CPU operations, this results in a considerable energy saving.

2.4.3.2 Radio Module

Figure 2.4 illustrates the large energy cost of transmitting data through a wireless interface. Processing power, however, does not require large energy contributions. Therefore, verifying the validity of the collected data can be crucial when operating on a limited energy budget. In a track and trace context, one could only transmit locations when motion is detected on the IoT node. Furthermore, the accuracy of the location can be verified on-node by means of parameters such as the dilution-of-precision (available on GNSS) modems prior to transmitting an inaccurate position.

Sending larger messages through an LPWAN communication link generally requires more time to be transmitted, thus spending more energy per message. The application payload size of one cycle should be kept at the minimum. It is inadvisable to add extra unnecessary static overhead bytes. When designing for one specific application, a byte numbered protocol could be put in place. An example is the Cayenne Low Power Payload format. Sensor types are allocated to predefined hex symbols in order to minimize the number of bytes. Despite the

convenience of such a payload format, defining your own dedicated format could achieve even lower payload sizes. Depending on the required range and accuracy of the sensor measurement, the number of bits per sensor reading can be reduced.

The energy consumption of IoT communication rises with higher payloads. However, due to protocol overheads, the energy cost per payload byte lowers when sending large payloads. Depending on the required response time, several sensor measurement intervals can be accumulated into a single message. This approach lowers the average energy cost per sensor measurement, as presented in [COv19] and investigated in Chapter 3.

2.4.3.3 Power Manager

As demonstrated in Equation (2.5), power consumption in CMOS circuitry can be divided in P_{switch} and P_{static}. When the controller is active, the power consumption is predominantly determined by P_{switch}. The supply voltage U_{DD} has a quadratic relation to the overall power consumed. When dealing with voltage sources significantly higher than the operating voltage, adding voltage regulation can yield significant power savings. Recently, buck-boost switching converters with ultra-low quiescent current have become available. Some even provide dynamic voltage scaling (e.g., Texas Instruments TPS63900, [23]): multiple user-selectable output voltages on a single voltage rail to deliver power while keeping the system at the minimum voltage required to operate efficiently.

Use-case application – Acoustic Event Detection

Consider the case where, in an urban setting, acoustic noise pollution needs to be monitored. The application needs to detect whether noise levels are above a certain threshold and, if possible, provide information about the nature of the sound, e.g., music, explosion, mass panic, etc. To that end, wireless, battery-powered IoT nodes equipped with a microphone are dispersed in the city.

In this particular application, a microphone with (adjustable) wake-on-sound functionality can lead to significant energy savings, even if it – in this wake-on-sound mode – consumes considerably more energy compared to one that is completely turned off. To explain this trade-off, we inspect how an IoT node equipped with a regular low-power microphone would handle the detection of acoustic events. The microcontroller would have to periodically wake up the microphone (and amplifiers) to record audio and process it, i.e., determine whether the recorded sound level is above a set threshold. To make sure that even short events are detected reliably, this process would have to take place, say, every second. Even with the microcontroller's RTC running, which is needed for the periodic wake-ups, the system consumes an order of magnitude less power in sleep, compared to a system that uses a microphone which is continuously in wake-on-sound mode (see Table 2.6). In this particular example we consider the Vesper VM1010, which can be configured to wake up when the sound level is over a certain threshold. The threshold can be set fixed or even adjusted on the fly [26]. The wake-on-sound operation mode avoids the need to record and process all sound sequences, which requires several orders of magnitude more power. Hence, it is still more energy-efficient to use a microphone in wake-on-sound mode in situations where acoustic events that sporadically exceed a certain threshold and may be short, need to be detected.

Table 2.6: Comparison of sensing mechanisms for the acoustic event detection use case. Power estimates based on EFM32 controller and VM1010 microphone.

(a) Power consumption in each mode.

Mode	CPU state	Power
sensor off	EM2	5 µW
wake-on sound	EM3	50 µW
sample	EM1	1 mW
process	EM0	10 mW

(b) Average power (with 10 ms sampling/processing).

Events/h	wake-on-sound	periodic wake-up
3600	159 µW	115 µW
1200	86 µW	115 µW
100	53 µW	115 µW
10	50 µW	115 µW

2.5 Conclusions

Low-power wide-area network technologies are optimized for low-power and remote connections. Networks are deployed operating both in unlicensed spectrum and in cellular licensed bands. Still, a meticulous design and disciplined operation of the IoT nodes is essential to provide them a long autonomy. Furthermore, in order to allow the deployment of IoT nodes on an increasingly larger scale, extending the battery autonomy of these nodes is imperative. Due to the high number of devices, the maintenance and environmental impact [BSD16; HPSA18] becomes non-negligible.

In this chapter, the manners in which long-range wide-area connections are supported and how they are implemented in LPWAN technologies were elaborated upon. These PHY and MAC schemes need to be accommodated with hardware and software tailored to the intended applications. Based on the characterization of the anatomy of an IoT node, the required hardware considerations to achieve a low-power operation are discussed. Besides a careful IoT node design, strategies to extend the operation time of the node is discussed. This primarily involves applying mechanisms in the firmware and requires a detailed knowledge of the underlying layers and components. We conclude with an illustrative example of a real-world application on how the proposed design principles and choices could increase the autonomy of the IoT node.

This chapter has provided an overview of the fundamental principles for designing energy-aware IoT nodes. The various methods for achieving energy efficiency have been explored, including optimizing node design and implementing energy-efficient wireless links. The concepts involving wireless links, are further evaluated in Chapter 3, where this PhD delves into a real-life experimental analysis of the energy consumption to assess the effectiveness of LoRaWAN, Sigfox, and NB-IoT for remote IoT use cases.

Chapter 3

Energy-Based Evaluation of LoRaWAN, Sigfox and NB-IoT

This chapter delves into an experimental examination of what parameters influence the energy consumption of IoT nodes and how they can be adjusted to minimize energy consumption, while in Chapter 2, the focus is on how energy-saving mechanics are designed into IoT nodes and in LPWANs.

A thorough evaluation is conducted of the three prominent LPWAN technologies that are the primary focus of this PhD, namely LoRaWAN, Sigfox, and NB-IoT. This comprehensive analysis is based on empirically obtained energy profiles. Each technology is evaluated in terms of the battery autonomy of the IoT node and the influence of network parameters on their respective battery autonomy.

Part of the content presented in this chapter is based on:

G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "An Experimental Evaluation of Energy Trade-Offs in Narrowband IoT," 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), New Orleans, LA, USA, 2020, pp. 1-5, doi: 10.1109/WF-IoT48130.2020.9221010. [LCVD20]

Contributions: The author conceptualized, and conducted the experiments presented in this chapter. The calculator tools and conclusions are also the sole

contribution of the author and have not been published elsewhere. In [LCVD20], on which this chapter is partly based on, the author took the lead in conceptualizing, conducting the experiments, and writing the paper.

3.1 LoRaWAN Evaluation

In this section, we examine and evaluate the energy consumption of a LoRaWAN modem in each individual state, creating a comprehensive energy model. Unlike cellular networks, where communication is typically initiated by the node but primarily controlled by the network, LoRaWAN allows the IoT node itself to drive and manage the communication process. As a result, the energy consumption of a LoRaWAN modem is highly customizable and can be optimized for the limitations of the use-case.

3.1.1 Energy Profile

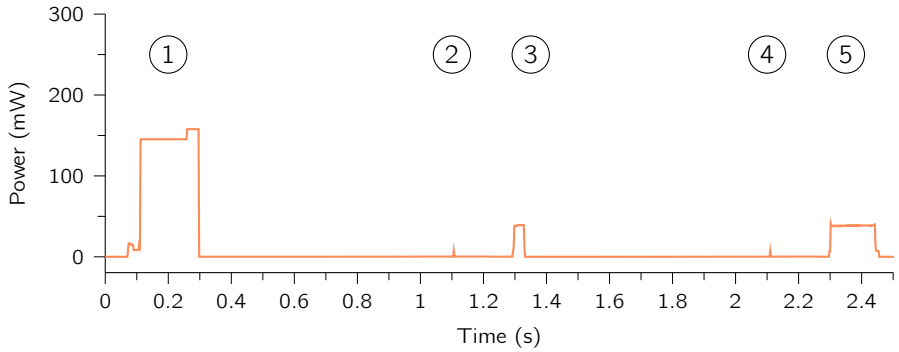


Figure 3.1: Measured power consumption of a LoRa node (at SF 12, output power of 14 dBm): (1) transmit, (2) processing, (3) first receive window, (4) processing, (5) second receive window.

A typical power profile of a LoRaWAN class A device is depicted in Figure 3.1. For these measurements the RF Solutions RF-LORA-868-SO LoRa transceiver is used, based on a Semtech SX1272. The LoRaWAN stack is running on a Silabs STK3400 "Happy Gecko" board, which features an ARM Cortex-M0+ based microcontroller.

The sequence of events with regard to sending LoRaWAN messages discussed in Section 2.3.1.2 can be recognized in the power profile. First, the IoT node

Table 3.1: Energy profile [33] of a LoRaWAN modem, operating in class A mode. For this model, the 863 MHz to 870 MHz frequency plan is selected, with SF 9 as data rate for the second receive window.

State no. on Figure 3.1	State Description	Power (mW)	Duration (ms)
-	Sleep	5.7e-3	-
1A	Transmit setup	12.5	40
1B	Transmission	Tab. 3.2	Eq. 3.1
2, 4	Processing	15	5
-	RX window 1 delay	5.7e-3	1000
-	RX window 2 delay	5.7e-3	$1000 - \text{len}\,(\text{RX window 1})$
3A, 5A	Receive setup	8.25	3.4
3B	Receive window 1	36.96	$\text{airtime}\,(\text{SF} = \text{SF}_{\text{TX}})$
5B	Receive window 2	34.65	$\text{airtime}\,(\text{SF} = 9)$

transmits a message (1), where the power consumed is dependent on the selected RF output power of the modem. The time it takes to transmit a single symbol is related to the selected data rate (e.g., spreading factor) and bandwidth BW. The time on air dependency on the spreading factor originates from the defined symbol duration T_{symb} in LoRaWAN:

$$T_{\text{symb}} = \frac{2^{\text{SF}}}{\text{BW}}. \tag{3.1}$$

After an uplink transmission, one, potentially two, receive windows are opened. The data rate and frequency band used for downlink data reception during the first receive window in LoRaWAN are closely tied to the data rate and frequency band employed during the uplink transmission. This relationship is documented in the relevant LoRaWAN regional specifications (e.g., [6]). On the other hand, the second receive window utilizes a fixed data rate and frequency band, which vary based on the network region and operator requirements.

The obtained energy profile is summarized in Table 3.1.

3.1.2 Factors Affecting Energy Consumption

Payload Length. The payload length has a direct influence on the airtime required to send a LoRaWAN message. The data in a LoRaWAN message consists of both header data (n_{overhead}) and application payload (n_{payload}), which together determine the total size (n_{total}) of the message:

$$n_{\text{total}} = n_{\text{overhead}} + n_{\text{payload}}. \tag{3.2}$$

The headers of a LoRaWAN message contain metadata such as the device address, frame counter, and CRC (Cyclic Redundancy Check) value. The total size of all headers in a LoRaWAN message typically ranges from 12 to 15 bytes, depending on whether MAC (Medium Access Control) commands are included.

The total number of symbols n_{sym} that are be transmitted is equal to [BKD+22]:

$$n_{\text{sym}} = 8 + \max\left(\text{ceil}\left(\frac{8n_{\text{total}} - 4\text{SF} + 28 + 16}{4\,(\text{SF} - 2\text{DE})}\right) \cdot (\text{CR} + 4)\,, 0\right). \tag{3.3}$$

In this equation, SF represents the spreading factor, DE indicates whether low data rate optimization is enabled (0 if disabled, 1 if enabled). This parameter is best enabled when using high SFs (SF 11 or 12), due to the long packet duration. The DE option adds a small overhead to increase robustness to frequency variation over the duration of the transmission [CMVG17]. CR represents the coding rate. The coding rate is denoted as a number between 1 and 4: a coding rate of 4/5 is indicated by $\text{CR} = 1$. Other coding rates include: 4/6 as indicated by $\text{CR} = 2$, 5/7 as indicated by $\text{CR} = 3$, and 4/8 as indicated by $\text{CR} = 4$. The time it takes to transmit a LoRaWAN message T_{air} is thus equal to:

$$T_{\text{air}} = T_{\text{symb}} \cdot n_{\text{symb}}. \tag{3.4}$$

Equation (3.3) (and subsequently Figure 3.2) reveal that due to interleaving and forward error correction in LoRaWAN, data is transmitted in fixed-length blocks. As a result, a small change in the payload size does not always imply a change in the airtime. The block length varies depending on the payload length, utilized spreading factor (SF), and bandwidth. For instance, when sending a message with a payload size ranging from 8 to 12 bytes, the time on air remains constant, as does the energy consumption, for a SF of 12 and a bandwidth of 125 kHz. This presents an opportunity to dynamically exploit this characteristic, enabling optimal utilization of the payload by filling it with useful data while considering the employed LoRaWAN parameters.

The relation between the energy consumption per payload byte and the length of the payload is depicted in Figure 3.3. As expected, the average energy consumption per payload byte decreases when sending larger packets. The fixed energy costs of sending a LoRaWAN message get distributed over more information. Consequently, to save energy, non-time-critical data can be accumulated, because by increasing

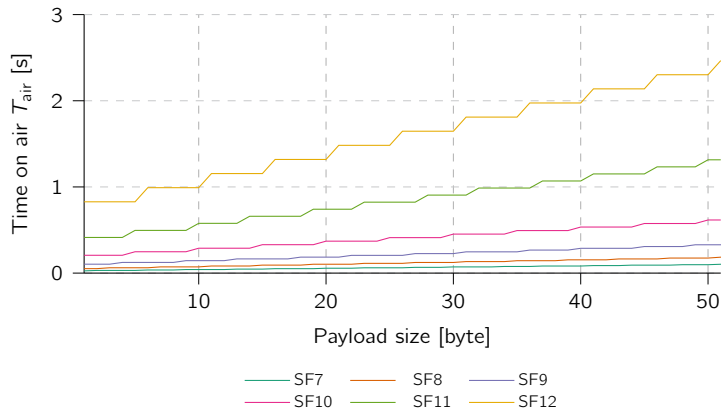


Figure 3.2: The total airtime it takes to transmit a LoRaWAN message, with respect to the payload size. A total header size of 12 byte is used: no MAC commands are included.

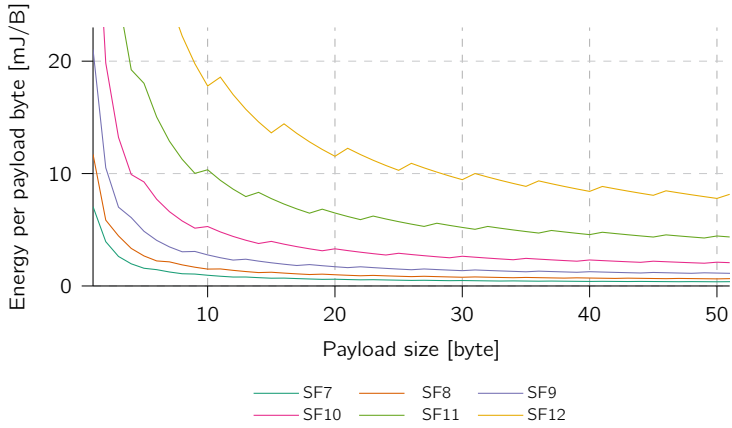


Figure 3.3: The total energy per payload byte in a single LoRaWAN message, versus the payload size, for various SFs (with a fixed *BW* of 125 kHz, coding rate 4/5, and transmit power of 14 dBm). Energy consumption derived from the energy profile obtained in Table 3.1.

the payload size (i) the overhead related to header information decreases, (ii) the overhead of starting and initializing a transmission lowers, (iii) the number of retransmissions in a stable propagation environment reduces, (iv) the number of down-link receive windows is also lower. The effect is especially significant for messages when sending small payload sizes.

Aggregating messages in LoRaWAN can slow down the adaptation process when using ADR. While the impact is negligible in static situations, it becomes more significant in dynamic scenarios. To adapt faster, IoT nodes can send a series of smaller packets initially, reducing airtime and energy for packets with non-optimal parameters. This allows for faster evaluation of different settings and identification of the optimal configuration. Once determined, the devices can switch to transmitting larger packets that contain aggregated data for improved energy efficiency.

Data Rate. Lower data rates (i.e., higher SF) result in higher energy consumption per transmitted bit due to the longer airtime required for transmission. The prolonged airtime necessitates that the device's radio remains active for extended periods, consuming more energy. In contrast, higher data rates (i.e., lower SF) enable quicker data transmission, leading to shorter airtime and reduced energy consumption per transmitted bit. As observed in Figure 3.3, increasing the SF by one step almost doubles the time on air (for the same bandwidth).

However, higher data rates impose limitations on the receiver's sensitivity (see Table 2.2 and Figure 2.8), which in turn affects the communication range. An optimal data rate (i.e., SF) for specific coverage levels can be determined. Nodes in close proximity to the receiving LoRaWAN gateway can likely transmit data successfully at the highest data rate (lowest SF). However, nodes located indoors or farther away from the gateway require transmission at a more energy-intensive, lower data rate (i.e., higher SF). This trade-off is optimized by enabling the ADR algorithm on the LoRaWAN IoT node (see Section 2.3.1.2).

Transmit Power. Table 3.2 provides an overview of the common output power selections for the LoRa modem mentioned earlier, along with the corresponding power consumption.

Increasing the transmit power enhances the communication range but comes at the expense of higher energy consumption. Thus, an optimal transmit power level exists for a given distance, balancing communication ability and energy efficiency. This ensures that the node can maintain reliable communication without excessively

Table 3.2: Measured power [33] (by the transmitter) for the transmit state, depending on the selected transmit output power. Note that according to ETSI regulations in the 868 MHz band, the option of 20 dBm output power can only be used in the 869.5 MHz channel [12].

Selected transmit power (dBm)	2	5	8	11	14	20
Power (mW)	91.8	95.9	101.6	120.8	146.5	481.5

draining its energy resources. To look for an optimal transmit power, a node tries to dynamically adjusted the transmit power value by using the LoRaWAN ADR algorithm.

In Table 3.2, it is evident that the energy consumption does not decrease proportionally with the selected transmit power. The power draw at lower selected transmit powers shows only a slight decrease. For instance, when comparing the energy reduction from 8 dBm to 5 dBm, the power consumption is only reduced by 5.6 %. However, when changing from 11 dBm to 8 dBm, the transmit power is lowered by 16 %. This behavior is primarily due to the interplay between the power amplifier (PA) gain and the overall energy consumption of the transceiver. Extensive studies by Cripps [Cri06] have documented the impact of PA gain on the energy efficiency of PAs in wireless communications. This is due to the lower efficiency of the PA when it works at a larger back-off.

It should be noted that some LoRa transceivers include an extra PA to boost the transmit power beyond 14 dBm. The use of this optional high-gain PA drastically increases energy consumption of the IoT node: a more than 3 times increase when compared to the 14 dBm output option. Moreover, according to regulations imposed by European telecommunications standards institute (ETSI) in the 868 MHz band, the option of 20 dBm output power can only be used in the 869.5 MHz channel [12].

Adaptive Data Rate. A LoRaWAN node can use ADR to adjust its SF and transmit power based on network conditions. As discussed in Section 2.3, ADR minimizes collisions and improves network capacity in LPWAN. It prioritizes selecting the lowest SF and transmit power for reliable communication and thus reducing the energy consumption.

Receive Windows. Adjusting the parameters of the receive windows that are opened after an uplink transmission, can lower energy consumption of a LoRaWAN node.

When a node detects downlink information in the first receive window, using the same SF as the uplink data frame, it skips opening the second receive window. This approach helps to conserve energy by avoiding unnecessary listening periods with a potentially higher SF. This is confirmed by Casals *et al.* [CMVG17], observing that requesting an acknowledgment can lower the overall energy consumption because the node does not open a second receive window after detecting a downlink message in the first window.

For the second receive window, the network settings determine a fixed SF. Semtech defines a default spreading factor of 12, but for energy conservation purposes, a lower SF would be more suitable [COv19]. For example, The Things Network proposes SF 9 as an alternative. Using a lower SF results in faster reception of downlink messages, leading to lower energy consumption at the node, yet needs to be supported by the network.

Another approach to conserve energy in downlink communication is to eliminate the opening of receive windows altogether. However, this would result in the node losing the capability to receive MAC-related commands, effectively disabling the ADR algorithm. To strike a balance between energy savings and maintaining necessary communication, one option is to selectively open receive windows only for a portion of the transmitted messages. This approach allows for downlink communication when required, enabling the node to receive vital MAC commands while minimizing energy expenditure. It's important to emphasize that this approach is experimental and not LoRaWAN compliant.

3.1.3 Energy-Based Evaluation

The power consumption of a LoRaWAN device can be reliably predicted based on the above-mentioned criteria (e.g., payload size and transmit power). In this section, the battery autonomy of a LoRaWAN IoT node is evaluated, based on the energy profile obtained in Section 3.1.1.

3.1.3.1 Battery Autonomy Calculator Tool

To accurately determine the autonomy of a LoRaWAN node on a single battery charge, a web-based calculator has been developed.[1] (see Figure 3.4).

The calculator comes with a preloaded energy profile, obtained from the data presented in Table 3.1 and Table 3.2. However, users also have the option to input

[1]The calculator can be accessed at dramco.be/tools/lorawan-battery.

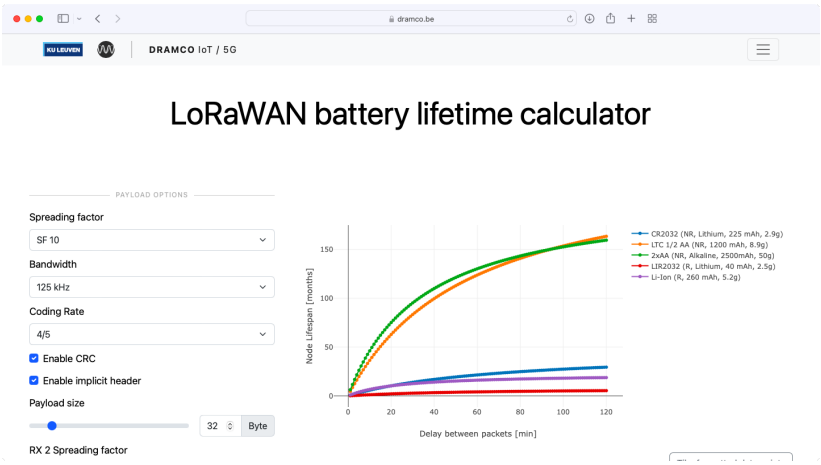


Figure 3.4: Screenshot of the presented LoRaWAN battery autonomy calculator tool. This tool can be used to accurately predict the autonomy of an IoT node, including the self-discharge rate of the selected battery.

a custom energy profile according to their specific requirements. The calculator tool calculates the expected autonomy of an IoT node, depending on how often data is being transmitted. It takes into account the LoRaWAN settings, which are associated with the factors that determine the energy consumption (discussed in Section 3.1.2), as well as use-case parameters and IoT node properties:

- LoRaWAN network parameters: spreading factor, bandwidth, coding rate, CRC enabled, implicit header option enabled, receive window settings
- Payload size
- Energy profile: sleep power consumption, transmit power consumption, receive power consumption, processing (and sensing) power consumption, processing (and sensing) duration
- Battery profile: energy capacity, self-discharge rate

3.1.3.2 Evaluation

The calculator tool presented above allows for the evaluation of all uplink-centric use cases described in Chapter 1. In this evaluation, the influence of the payload size, SF, and transmit power selection (and thus the importance of ADR) is examined.

First, a baseline power consumption is considered. In the case of environmental monitoring, which for example requires sending a 10-byte payload every 10 minutes[2], a battery life of 5.2 years can be achieved using a commonly available double, non-rechargeable, AA Alkaline battery pack with a combined capacity of 2500 milliampere-hours (mAh) (as depicted in Figure 3.5), including the self-discharge rate of 3 % per year [18].

Payload Size. To extend the autonomy of the sensor node, the strategy of aggregating sensor values can be employed. By transmitting a payload of 40 bytes every 40 minutes, for example, the autonomy can be extended to 8.4 years, effectively prolonging the autonomy of the sensor node by 61 % (see Figure 3.5). By further aggregating data, additional energy savings can be attained.

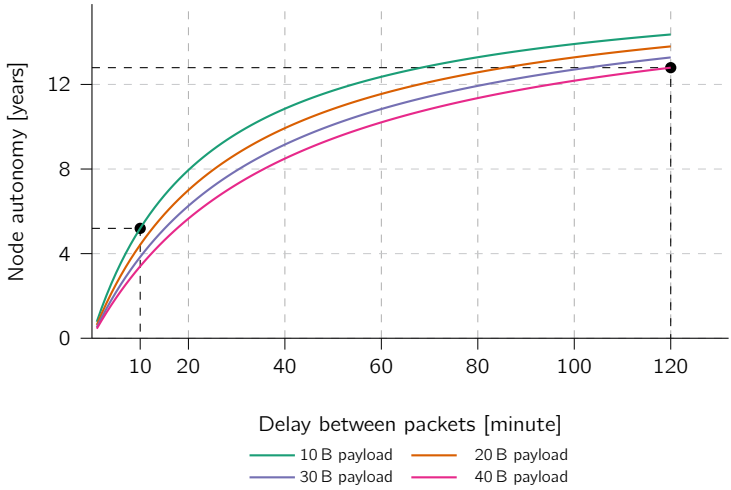


Figure 3.5: Battery life when sending 10 B and 40 B payloads with a LoRaWAN IoT node. Network parameters used in this calculation: SF 10, bandwidth of 125 kHz, transmit power of 14 dBm, coding rate of 4/5, and the second receive window is set to SF 9.

Spreading Factor and Transmit Power. The battery autonomy of an IoT node can be significantly influenced by network conditions. The most significant increase in battery autonomy can be achieved by lowering the SF, as observed in Figure 3.6. Comparing a node transmitting on SF 7 to a node transmitting on SF 12 effectively doubles the autonomy of the IoT node.

[2]The 10-byte payload is sent using SF 10, a bandwidth of 125 kHz, a transmit power of 14 dBm, a coding rate of 4/5, and the second receive window is set to SF 9.

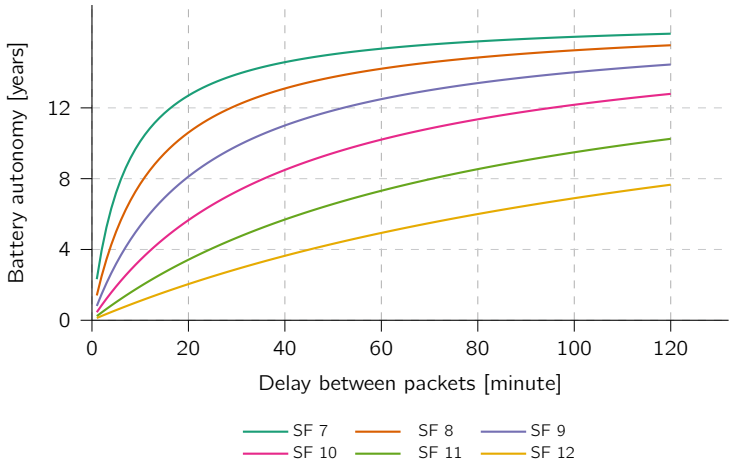


Figure 3.6: Battery life when sending 40 B payloads with a LoRaWAN IoT node with various spreading factors. Parameters used in this calculation: 40 B payload, bandwidth of 125 kHz, transmit power of 14 dBm, and coding rate of 4/5, the second receive window is transmitted in SF 9.

The transmit power also has an impact on the autonomy of a LoRaWAN node and is illustrated in Figure 3.7. When comparing a lower transmit power of 2 dBm to a higher transmit power of 14 dBm, the autonomy of the node, which encompasses other energy consumption sources such as sleep and receive windows, only improves by 8 %. Since the option of using 20 dBm transmit power is specific to a single band and not standard in LoRaWAN, it is omitted from this evaluation.

It is evident that optimizing the battery autonomy relies on selecting the appropriate SF and transmit power. This underscores the importance of enabling ADR for static IoT nodes, wherein the node dynamically adjusts its SF and transmit power for optimal performance. Enabling ADR incurs minimal costs, primarily associated with receiving downlink messages, which are significantly outweighed by the potential gains in battery autonomy. It should be noted, however, that the ADR algorithm exhibits a slow response to dynamic changes in channel conditions. Consequently, enabling ADR is only a valid approach for stationary nodes with static channel conditions.

Receive window. The choice of SF for the second receive window has minimal impact on the battery autonomy of an uplink-centric IoT node, as illustrated in Figure 3.8. Shifting from SF 12 to SF 9 as the data rate for the second receive window results in a modest extension of the battery life by approximately 3.2 %.

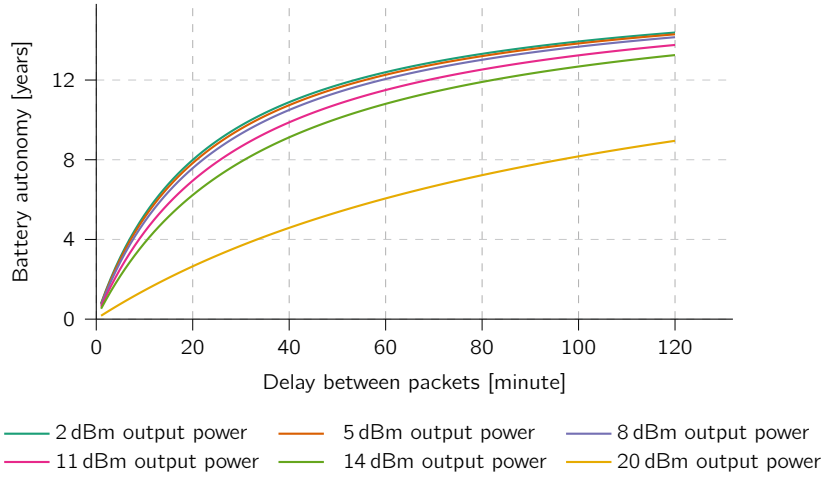


Figure 3.7: Battery life when sending 40 B payloads with a LoRaWAN IoT node with various transmit power settings. These parameters were used in this calculation: 40 B payload, SF 10, bandwidth of 125 kHz, and coding rate of 4/5, the second receive window is transmitted in SF 9.

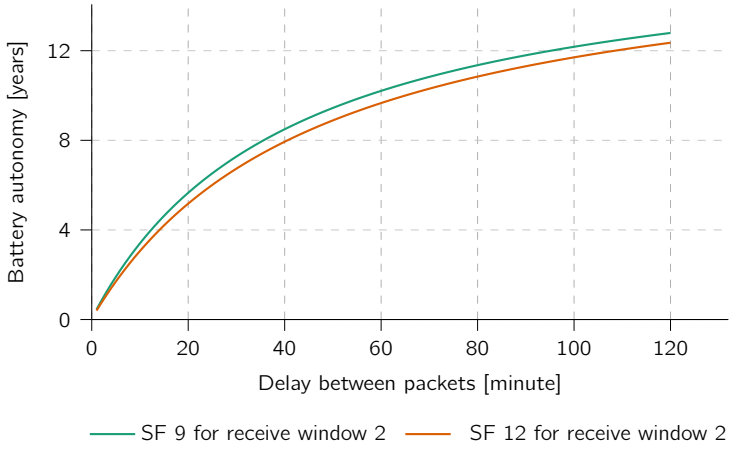


Figure 3.8: Battery life when sending 40 B payloads with a LoRaWAN IoT node with respect to the SF of the second receive window. Parameters used in this calculation: 40 B payload, SF 10, transmit power of 14 dBm, bandwidth of 125 kHz, and coding rate of 4/5.

3.2 Sigfox Evaluation

3.2.1 Energy Profile

A typical power profile of a Sigfox node is depicted in Figure 3.9. For these measurements, a custom developed module based on the Murata CMWX1ZZABZ-091 is used. This is a Sigfox capable modem powered by an ST ARM Cortex-M0+ microcontroller and an embedded Semtech SX1276 transceiver. Notably, the user-programmable microcontroller is a distinguishing feature of this module, enabling energy savings by eliminating the need for an additional microcontroller to drive the desired use case.

The power profile of a Sigfox node exhibits three distinct states associated with bidirectional communication: (1) main uplink communication, consisting of three repetitions, (2) opening of a receive window, and (3) confirmation of the received downlink message. It's important to note that bidirectional communication in Sigfox is optional, so states (2) and (3) are only relevant when downlink communication is required, such as for acknowledgments or transferring sensor configuration changes.

For a more detailed analysis of the main uplink communication, refer to Figure 3.10. This figure provides a comprehensive view of the power profile, clearly illustrating the repetition of the uplink message as described in Section 2.3. During the main uplink communication, the IoT node awakens from deep sleep and prepares for message transmission (1A). It starts the transmission of the first uplink frame

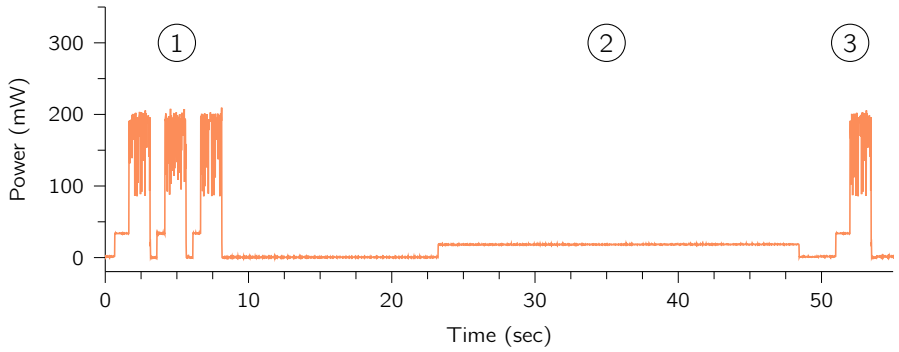


Figure 3.9: Measured power consumption of a Sigfox node (transmitting a 4-byte payload with an output power of 14 dBm): (1) three transmission frames, (2) receive window for downlink communication, (3) downlink acknowledgment.

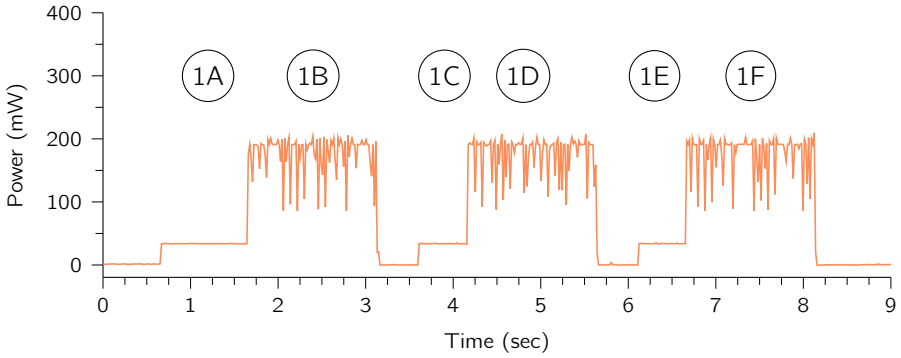


Figure 3.10: Detail of the power consumption of a Sigfox node sending 4 byte uplink data (at 14 dBm output power).

(1B), with the power consumption at this stage dependent on the chosen RF output power of the modem. The time required to transmit a single bit is determined by the operating region. For instance, in Europe or the USA, Sigfox radio configuration (RC) 1 is used, which defines a bit rate of 100 bit/s. After the transmission, the node enters a predefined sleep interval before waking up to start the second uplink frame. Typically, this sequence is repeated for a total of three uplink frames, which collectively form a single Sigfox uplink message.

Once the three data frames have been sent, the node enters a deep sleep state. However, if downlink communication is required, the node wakes up after a predefined interval to open a receive window. Upon receiving the downlink message (or after a predefined timeout), the node closes the receive window and sends an uplink confirmation message to acknowledge the successful reception of the downlink message.

The obtained energy profile is summarized in Table 3.3.

3.2.2 Factors Affecting Energy Consumption

Payload Size. The payload length has a direct influence on the energy consumption of a Sigfox node, through the time it takes to transmit a single Sigfox uplink frame T_{TX} [GVV+19]:

$$T_{\mathrm{TX}} = 8 \cdot \frac{n_{\mathrm{total}}}{\mathrm{BR}_{\mathrm{UL,bit}}}. \tag{3.5}$$

Table 3.3: Energy profile of a Sigfox modem. For this model, European regional parameters are used (e.g., RC 1).

State no. on Figure 3.9	State Description	Power (mW)	Duration (s)
-	Sleep	$34.6e-3$	-
1A, 3A	First transmit setup[a]	34.07	0.98
1B, 1D, 1F, 3B	Transmission (14 dBm)	152.65	Eq. 3.5
-	TX delay	$34.6e-3$	0.43
1C, 1E	Consecutive transmit setup[a]	34.26	0.544
-	RX window delay	$34.6e-3$	15
2	RX window	19.63	max: 25
-	ACK delay	$34.6e-3$	2.5

[a] Other Sigfox modems do not necessarily demonstrate a first transmit setup phase (1A) or any intermediate setup phases (1C, 1E).

For applications in Europe or the USA, $BR_{UL,bit}$ is defined as 100 bit/s.

In the Sigfox protocol, messages can have a payload length ranging from 0 to 12 bytes. The energy efficiency per useful payload byte, as illustrated in Figure 3.11, exhibits an expected trend: as the packet size increases, energy consumption per payload byte decreases. To optimize energy usage, small non-time-critical data can be accumulated and sent in a single Sigfox transmission. Note that energy savings are limited by the 12-byte payload restriction of the Sigfox protocol.

Not all payload lengths originating from the use case directly affect the effective transmitted payload length and subsequent energy consumption, hence the stepped gradient in Figure 3.11. Sigfox incorporates payload padding techniques to accommodate different payload lengths. Payload padding involves adding extra bytes to the message for specific payload lengths, as depicted in Table 3.4. From a use-case perspective, this provides the opportunity to leverage the additional bytes added through payload padding for transmitting supplementary and relevant information without added energy cost.

Note that, downlink messages also employ a fixed length of 8 bytes. Extra padding is added when less information is to be transmitted.

Transmit Power. The output power of the Sigfox node directly affects its energy consumption. Table 3.5 provides an overview of the common power selections for Sigfox, along with the corresponding power consumption for the

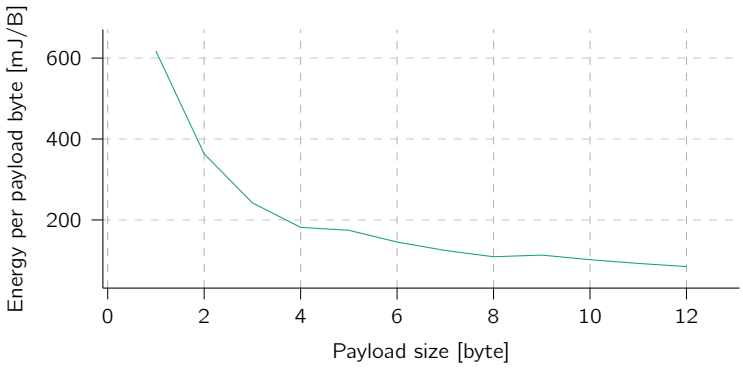


Figure 3.11: The total energy per payload byte in a Sigfox message, versus the payload size (transmitted at 14 dBm without bidirectional communication, an overhead size of 14 B is used). Note the stepped gradient, a result of payload padding employed in Sigfox.

Table 3.4: Payload padding in Sigfox: relation between the payload length originating from the use-case, the effective transmitted payload length, and the frame transmission time.

Payload length (use case) [B]	Effective payload length n_{payload} [B]	Frame transmission time T_{TX}[s]
<1 bit	0	1.12
2 bits - 1 byte	1	1.2
2 - 4 bytes	4	1.44
5 - 8 bytes	8	1.76
9 - 12 bytes	12	2.08

Sigfox modem TD1204[3]. The results demonstrate that reducing the transmit power from 14 dBm to 0 dBm, lowers the power consumption during the transmit phase by 43 %.

Determining the optimal transmit power for a specific location heavily relies on the Sigfox network coverage. In urban areas with dense coverage, lower transmit powers can be used for fixed Sigfox nodes while still achieving reliable communication. Although the output power can be changed through downlink communication, Sigfox does not implement an automatic algorithm to adjust the transmit power.

[3]Not all Sigfox modems support transmitting messages at RF output powers lower than the maximum transmit power of 14 dBm (in Europe). For instance, the Murata CMWX1ZZABZ-091 modem used in Section 3.2.1 does not allow setting a custom transmit power below 14 dBm. On the other hand, modems like the Telecom Design TD1204 Modems do support lowering the transmit power, but they are typically preassembled with an RF matching network optimized for 14 dBm output power.

Table 3.5: Measured power (for the Telecom Design TD 1204 Sigfox modem) for transmit power state in Sigfox, depending on the selected transmit output power.

Selected transmit power (dBm)	0	5	8	10	12	14
Power (mW)	89.2	93.7	111.1	126.1	141.57	156.09

It is worth mentioning that Sigfox classifies its certified nodes into three categories: U0, U1, and U2, based on their RF radiated performance, ranging from 14 dBm to 0 dBm respectively. However, this classification does not necessarily indicate that U2 devices consume less energy. For instance, a device with an output power of 14 dBm combined with a suboptimal antenna design can fall into the U1 or U2 device class. Conversely, the transceiver output power can also be set at the factory level for energy-saving purposes. In such cases, a U1 or U2 class device does not mean that the hardware used is incapable of achieving U0 performance.

Downlink Messaging. Bidirectional communication is possible in Sigfox, although not mandatory. As shown in Figure 3.9, a significant amount of time and energy is dedicated to waiting for downlink communication. Without acknowledgment, 58 % of the energy in a bidirectional communication is consumed during the downlink phase, which increases to 75 % with acknowledgment. To optimize energy usage, it is recommended to restrict the number of downlink messages for a node to meet the requirements of the use case. A practical approach could involve allowing a Sigfox node to listen for downlink messages once per day, primarily for potential configuration changes, while transmitting sensor data every hour. This approach aligns with Sigfox's network plans, where downlink messaging is restricted to a maximum of four messages per day, whereas uplink messages can reach up to 140 per day, providing flexibility for transmitting sensor data.

Furthermore, to reduce energy consumption during the receive window, the Sigfox Library version 2.10 incorporates discontinuous reception techniques such as preamble sampling [11], although their practical performance has not been tested yet.

3.2.3 Energy-Based Evaluation

This section focuses on assessing the battery life of the Sigfox node based on the energy profile obtained in Section 3.2.1. The evaluation methods employed

are akin to the energy-based evaluation methods utilized in the assessment of LoRaWAN (Section 3.1.3.2).

3.2.3.1 Battery Autonomy Calculator Tool

The calculator tool introduced in Section 3.1.3.1 has been specifically tailored for Sigfox.[4] This adapted calculator takes into account the energy profile outlined in Table 3.3 and Table 3.5. The calculator inputs have been modified to align with the Sigfox protocol and include the following parameters:

- Sigfox parameters: data rate, number of downlink opportunities, downlink acknowledgment, number of Sigfox frames
- Payload size
- Energy profile: sleep power consumption, average transmit power consumption, receive power consumption, processing power consumption, processing duration
- Battery profile: energy capacity, self-discharge rate

3.2.3.2 Evaluation

The Sigfox calculator tool facilitates the assessment of various uplink-centric use cases outlined in Chapter 1. Specifically, the same application use-case as evaluated in the LoRaWAN assessment in Section 3.1.3.2 is considered as baseline operation, where a 10-byte payload is transmitted using Sigfox every 10 minutes[5]. By utilizing a double AA Alkaline battery with a combined capacity of 2500 mAh, a battery life of 5 months can now be achieved (as illustrated in Figure 3.12).

Payload Size. In the example mentioned above, the 10-byte payload cannot be included twice in a single Sigfox message, due to the limited payload size. Yet, to illustrate the impact of data aggregation, another case with smaller sensor data of 2 bytes is considered. For this case, it is feasible to include six data points within a single message and transmit it once every hour, instead of sending 2 bytes every 10 minutes. This approach leverages payload padding (see Table 3.4), providing

[4]The calculator adapted to the Sigfox protocol can be accessed at dramco.be/tools/sigfox-battery.

[5]The 10-byte payload is sent using a bit rate of 100 bit/s, with no downlink opportunity, and by sending out three Sigfox frames (thus two repetitions).

an opportunity to prolong the autonomy of the Sigfox node from 7 months (for sending out a 2-byte payload, every 10 minutes) to 2.8 years (for sending out a 12-byte payload, every hour) (see Figure 3.12).

It's worth noting that Sigfox allows for transmitting payloads of 0 bytes or even a single bit at a minimal energy consumption. This minimizes the frame transmission time T_{TX} to the absolute minimum of 1.12 s, which constitutes in energy savings of 6 % when compared to sending a single payload byte.

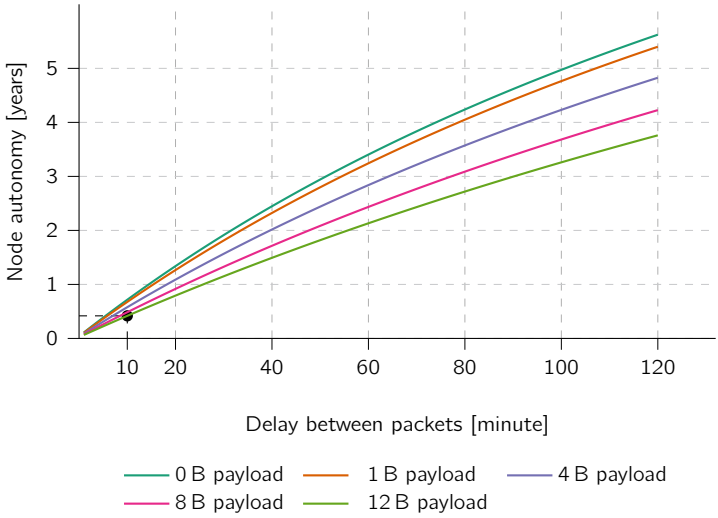


Figure 3.12: Battery autonomy when sending Sigfox messages with varying payload sizes. A transmit power of 14 dBm is used, while only considering the energy spent by sending uplink data.

Transmit Power. In areas with good network coverage, particularly in a dense Sigfox network, the transmit power used by the node can possibly be lowered. The effect of reducing the transmit power is demonstrated in Figure 3.13, where lowering the transmit power from 14 dBm to 0 dBm can extend the autonomy of the Sigfox node by 42 %. Yet, the energy savings achieved by lowering the transmit power to 5 dBm compared to 0 dBm are limited, though it does contribute to improved coverage within the Sigfox network. By utilizing 5 dBm, the autonomy of the node can be extended by 39 %.

Downlink Messaging. The impact of downlink communication on the autonomy of a Sigfox node is illustrated in Figure 3.14. Sigfox imposes a limit of at most four

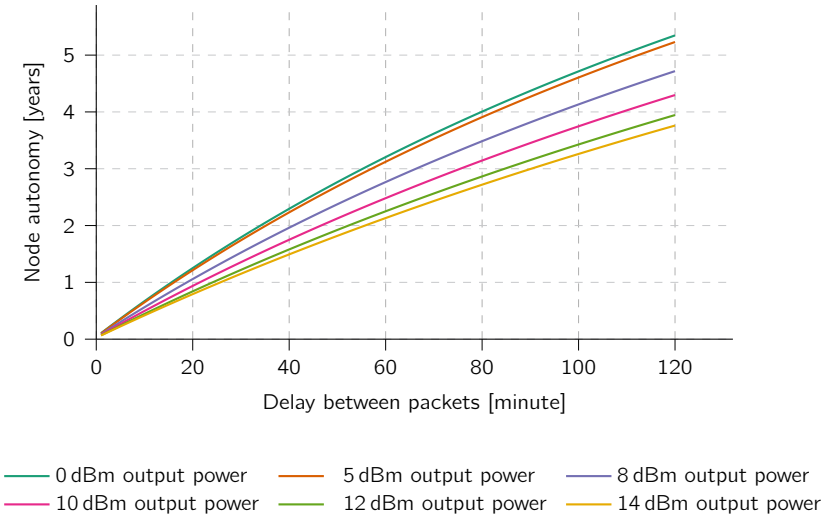


Figure 3.13: Battery autonomy when sending Sigfox messages with varying selections of transmit power. A payload size of 12 B is used, while only considering the energy spent by sending uplink data.

downlink messages per day, which translates to one message every six hours. When a Sigfox node checks for downlink messages only four times a day without employing downlink acknowledgment, its autonomy decreases by 5.9 % compared to the scenario where it never opens a receive window to check for downlink messages. In contrast, when downlink acknowledgment is employed, node autonomy decreases by 12.5 %.

If a node checks for downlink messages more frequently than necessary (for instance, when it requires latency-critical downlink communication), the autonomy of the node declines significantly. For instance, a node that checks for downlink messages every hour (without sending an acknowledgment) experiences a 27 % reduction of its autonomy.

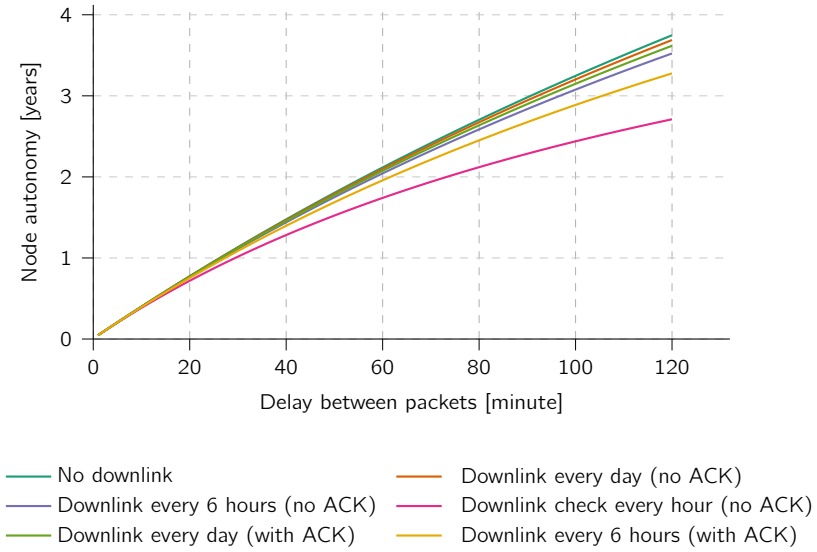


Figure 3.14: Battery autonomy when sending Sigfox messages with varying options regarding receiving downlink messages. An uplink payload size of 12 B is used, and a transmit power of 14 dBm is selected.

3.3 NB-IoT Evaluation

The adoption of cellular IoT technologies for communication can have a significant impact on the energy consumption of IoT devices, as demonstrated in [MMVP15]. In this section we derive a typical power profile of a NB-IoT node and examine the energy impact of several techniques introduced in NB-IoT.

3.3.1 Energy Profile

This assessment includes the effect of the recently deployed NB-IoT network. More specifically, the energy consumed by a NB-IoT modem for each individual state is measured and is depicted in Figure 3.15. As the majority of parameters impacting the energy consumption is defined by the network, this energy profile could yield different results than presented in literature. For example, Andres-Maldonado *et al.* [AAP+17] and Kovács *et al.* [KML+17] utilize the power consumption reported by 3GPP in [9]. The reported power consumption is evaluated by comparing it to our experiments, demonstrating that the advertised power consumption [9] is only valid for best-case scenarios as observed in our measurement campaign.

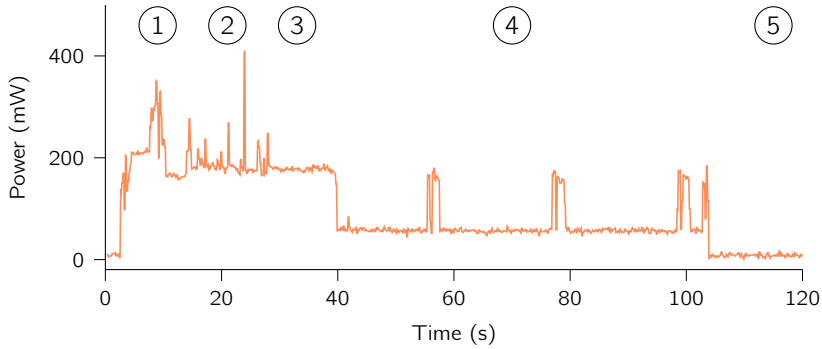


Figure 3.15: Measured power consumption of a NB-IoT node: (1) network search and join, (2) package sending, (3) connected mode DRX, (4) extended discontinuous reception mode, (5) power saving mode

An underestimate of the real energy consumption could potentially have a large impact on the design of IoT devices and algorithms.

A typical power consumption measurement is depicted in Figure 3.15. A schematic overview of modem states is presented in Figure 3.16. Several states can be recognized: (1) network search and join, (2) package send (3) connected mode discontinuous reception mode (DRX), (4) extended discontinuous reception mode and (5) power saving mode .

During (1), the modem boots and attempts to connect to the network, negotiating several network parameters (e.g., CE level, timer values, etc.). After the node has successfully joined the network, the package is sent (2). In this example, a payload of five random bytes was chosen. The payload is encapsulated in a user datagram protocol (UDP) packet, resulting in a total of 23 B. This represents a typical payload for machine-type communication (MTC). After the message is sent, the modem enters connected mode DRX (cDRX) (3). The radio circuitry stays active for a predetermined period, so downlink messages can be received, i.e., paging. If eDRX (4) is supported by both the network provider and the node, the modem is able to support periodical downlink communication. To conserve energy, the modem is put to sleep between downlink windows: paging time window (PTW). When the eDRX timer expires, the module enters PSM (5). PSM is the lowest power setting possible in NB-IoT. The modem is put in deep sleep and is not available to the network. The IoT node can periodically wake the modem, or let the specified PSM timer run out before reconnecting to the network.

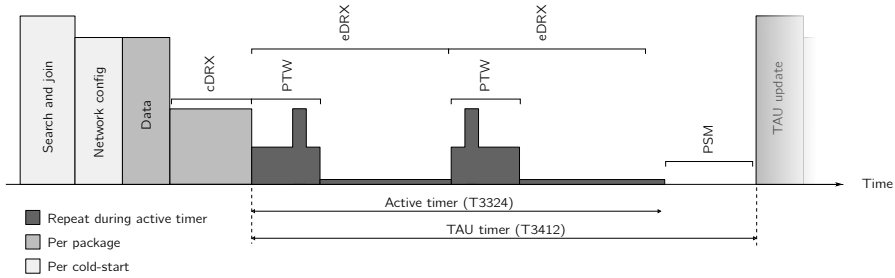


Figure 3.16: Schematic overview of NB-IoT modem states regarding power consumption. Several states are repeated per cold-start, per package and indefinitely repeated during the active timer.

Table 3.6: Overview of configurable parameters in NB-IoT and their impact on energy, availability and coverage. The columns *req. by* and *contr. by* describe if the user equipment (UE) or the evolved Node B (eNB), respectively, requests or controls the parameters.

Parameter	Req. by	Contr. by	Min.	Max.	Default	Affecting (when ↑)		
						En.	Range	Avail.
cDRX RRC Inactivity timer	-	eNB	0 s	60 s	20 s	↑	-	↑
eDRX Active Timer (T3324)	UE	eNB	0 s	186 min	Off	↓	-	↓
PSM TAU Timer (T3412)	UE	eNB	10 min	413 d	Off	↓	-	↓
CE Level	eNB	UE	0	2	0	↑	↑	-

3.3.2 Factors Affecting Energy Usage

The protocol procedures of NB-IoT have been thoroughly described in literature [8; RVMG16; RMG15]. Several new enhancements in 3GPP Release 13 stand out in respect of energy consumption of an IoT node. Here we focus on determining the impact of various low-power enhancements in NB-IoT in a practical deployment. These enhancements, in a practical setting, are described below.

Coverage Extensions. Three coverage enhancement (CE) levels have been specified in NB-IoT. This enables providers to ensure network connectivity even in hard to reach places such as underground parking lots.

The utilized CE levels are selected based on the quality of the wireless channel, estimated by the reference signals received power (RSRP). During the network

random access procedure, two RSRP thresholds are configured. Therefore, three CE levels can be configured: CE level 0 (good signal quality), 1 (mediocre signal quality) and 2 (poor signal quality). If there is poor network reception when establishing a connection, the CE level is increased to level 1 or 2. Several measures are enabled with each increase of the CE level, yet the most coverage gains savings are obtained from increasing repetitions to up to 128 [LSW+17; 8]. According to a large scale measurement campaign by T-Mobile [22], no coverage extension was required when measuring outdoors for 93 % of measurements. When measuring indoors, CE level 1 is used 27 %, while CE level 2 is used 19 % of the time. This demonstrates that T-Mobile provides a good coverage. Note that this depends on the operator, highlighting the need for these experiments.

Each increase in CE level has an energy penalty. Increasing the amount of repetitions increases the airtime and thus the energy consumption.

Paging and eDRX. When an idle device monitors the network for information, it consumes energy. Controlling paging is thus a crucial step into lowering the energy consumption of an NB-IoT node. Several steps have been taken in NB-IoT to limit the amount of paging. After a data packet has been sent, the UE first enters radio resource connection (RRC) connected state, followed by RRC idle state. The time a UE spends in RRC connected state is negotiated with the network. In connected state, the periodicity of a DRX cycle can be configured to be at most 10.24 s [16]. A DRX cycle consists of two phases. First, the UE monitors the network for a short period. Second, the UE stops monitoring the network for a longer time. To save energy, an extended form of DRX is introduced in NB-IoT: eDRX. When in idle state, a DRX cycle can be extended to up to 2 h 54 min 46 s. The node is not reachable by the network before the next eDRX cycle. Longer eDRX cycles result in lower response time but save energy.

PSM. PSM has been introduced in NB-IoT to allow for a node to go into deep sleep. In PSM, a device uses the least amount of power possible, requiring only to leave its RTC on. Devices in PSM are not reachable, but are still registered. The UE remains in deep sleep mode until a network transaction is needed; i.e., sending uplink data or expiration of the TAU timer. When sending uplink data, the UE wakes from PSM early and resumes its connection with the network to send data. The TAU timer keeps track of the time spent in PSM. The value of TAU is negotiated with the network before entering PSM. When the TAU timer expires, the UE wakes and resumes its connection with the network, checking available downlink messages. In NB-IoT, the maximum TAU value is 413 days.

Table 3.7: Overview of configured parameters used in the measurement campaign.

Parameter	Value	Min.	Max.
Packet protocol	UDP	-	-
Payload size	5 B	1 B	1460 B
eDRX cycle	20.48 s	10.24 s	2 h, 54 min
paging time window (PTW)	2.56 s	1.28 s	20.48 s
Active Timer (T3324)	60 s	0 s	186 min
Inactive Timer	10 s	0 s	60 s

3.3.3 Experimental Assessment

The impact of the aforementioned energy saving modes, in a realistic setup, is here studied. The experimental approach allows us to derive an energy profile of a NB-IoT node in a commercial network and hence also study the practical limitations imposed by the network.

3.3.3.1 Measurement Setup

To facilitate power measurements, a custom NB-IoT module was developed, hosting a Quectel BG96. The BG96 series are marketed as ultra low-power and thus ideal for IoT use cases. The modem supports both NB-IoT and GPRS. An integrated GPS receiver can be used for track-and-trace applications. The Proximus network is used for these tests.

The setup allows to measure the power consumption of the NB-IoT module and associate it with the current state of the modem. By doing so, a fine-grained energy model of the modem in each possible state is derived (combining Figure 3.15 and Figure 3.16), without the consumption of the peripheral hardware. To force the modem into a specific CE level, RF shielding is applied to the setup. In this manner, we obtain the energy consumption of each distinct state with different link qualities. Each packet consists of a five byte payload. The payload is encapsulated in a UDP packet, creating a 23 byte packet. In this experiment 132 power-cycles were measured with varying link quality. A power-cycle (Figure 3.16) consist of states only required in the case of a cold start (e.g., network configuration) and data communication related states (e.g., paging) which are periodic. The utilized configuration is shown in Table 3.7.

Table 3.8: Overview of the power consumption in different states with different CE levels. It demonstrates that measured power consumption is drastically higher than advertized by 3GPP [9] for all coverage levels.

Operation	Assumed Power (mW) [9]	Measured Average Power (mW)		
		CE level 0	CE level 1	CE level 2
Transmission (Tx)	545	408.0	843.0	1253.0
Reception (Rx)	90	182.0[a]	182.0[a]	182.0[a]
Sleep	3	7.1[a]	7.1[a]	7.1[a]

[a] As the CE level does not change reception or sleep average power, these values are constant for all CE levels.

3.3.3.2 Measured versus theoretical power profile

The parameters used in the measurement campaign are summarized in Table 3.7. The power consumption assumptions for NB-IoT reported by 3GPP [9], and commonly used in literature, and our experimental results are summarized in Table 3.8. It clearly illustrates that these assumptions underestimate the power consumption of commercial NB-IoT modems operating in actual deployed networks. This is especially significant in CE level 2 environments consuming twice the power mentioned in [9]. Depending on the provided coverage of the network operator, this could greatly reduce the expected autonomy of the devices.

3.3.3.3 Energy consumption w.r.t. received signal strength

The total energy required to transmit a 5-byte payload in NB-IoT as a function of the observed received signal strength is shown in Figure 3.17. As can be observed, the variance of the required energy increases with a decrease in received signal power. This indicates that in harsh propagation environments the energy consumption of the NB-IoT node is more difficult to predict. Furthermore, the energy increases exponentially with a decrease in signal strength.

The increase in energy consumption can be primarily attributed to two main factors: message repetitions and variable transmission power, along with the shift to a less energy-efficient modulation scheme (i.e., BPSK in CE level 2). These settings are subject to negotiation between the UE and the network and are part of the CE level setting. In scenarios with poor coverage, retransmissions become more frequent and are entirely depend on the state of the network (e.g., deployment, configuration, network load). This further exacerbates power consumption and contributes to an increase in energy usage variability. It's essential to note that

network coverage is determined by the network operator through meticulous deployment planning. Factors such as urban obstructions, hilly terrain, or large distances can significantly affect propagation, demanding strategic planning of base stations. These factors hamper an accurate predicting of the energy expenditure for NB-IoT nodes.

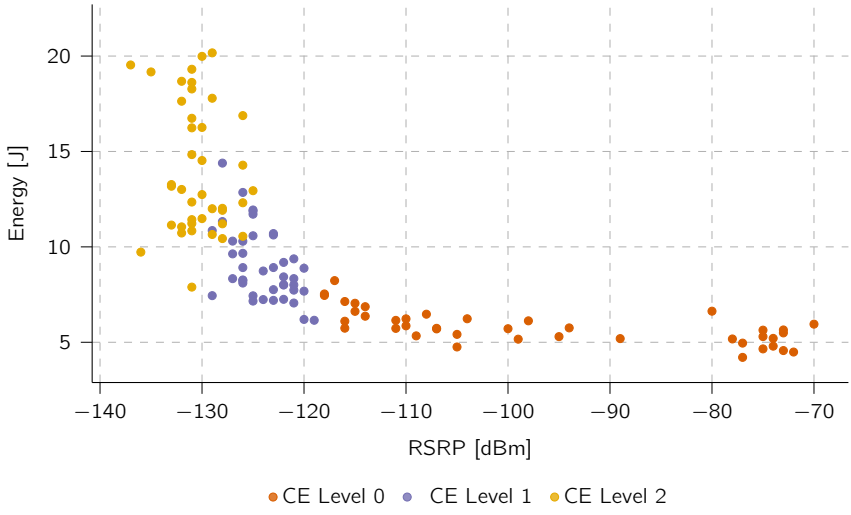


Figure 3.17: Total energy consumed when sending a 5-byte payload (18 byte overhead) with respect to the observed received power by the node, i.e., RSRP. The data points are grouped based on the coverage enhancement Level.

3.3.3.4 Energy share of each state under different link qualities

Figure 3.18 illustrates the effect of different CE-levels on the average energy consumption as well as the relative energy consumption of each distinct state. The join and network procedures require more energy with increased CE level; relative to the total energy. Fortunately, these states are only present in case of a cold-start. The total energy consumption in the remaining states, which contribute to the energy share of a packet transmission, decreases with the CE level. Hence, more energy is spent in networking than in transmitting the message. This is mainly due to joining the network, where the energy consumption increases with increased CE level. This implies that if the received power at the UE is low, we should opt for keeping the modem on, thereby omitting the need to re-connect. However, in other scenarios it is better to fully power-off the modem to optimize for power efficiency.

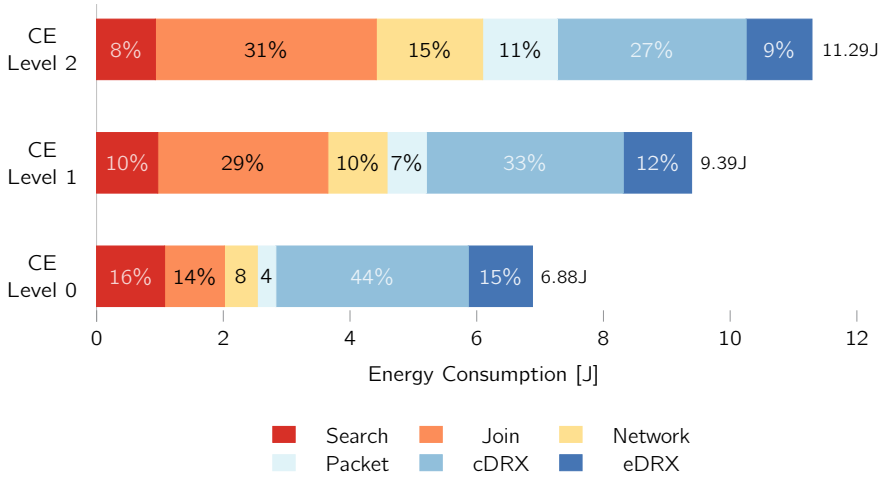


Figure 3.18: Average energy consumption of different NB-IoT modem states depending on coverage enhancement Level. The utilized configurations during the measurements is summarized in Table 3.7.

3.4 Conclusions

This chapter established the energy profile of LoRaWAN, Sigfox, and NB-IoT by evaluating their autonomy in the context of IoT nodes. The results, supported by empirical data, demonstrate that a typical IoT node designed for sporadic uplink communications can achieve an autonomy of over 10 years. However, this value is heavily influenced by various factors with respect to the use case such as the required payload size, and acceptable latency.

For LoRaWAN and Sigfox, calculator tools have been developed to estimate node autonomy based on input parameters that impact energy consumption. On the other hand, IoT nodes connected to the NB-IoT network are highly dependent on network configuration and conditions, making it challenging to accurately predict their energy consumption. As a result, no calculator tool for NB-IoT has been developed. Yet, the obtained energy profile provides valuable insight in the energy consumption of NB-IoT.

The established energy profiles are inherently specific to the employed hardware implementation. However, in the conducted work, special attention has been paid to selecting the most energy-efficient implementation possible at that time. Opting for alternative hardware, such as a different NB-IoT modem [MABV19] or Sigfox transceiver [GVV+19], may result in varying absolute battery autonomy due

to factors like slightly different transmitter (TX) or sleep current consumption. Nonetheless, the general conclusions based on a comparative analysis of the impact of various parameters across the studied LPWAN technologies remains valid and holds true throughout this chapter, regardless of hardware implementation. Furthermore, the calculator tools designed for LoRaWAN and Sigfox enable the input of precise current measurements to create a custom energy profile for a different hardware implementation. This capability ensures accurate predictions of the battery autonomy for customized nodes.

The work presented in this chapter offers an energy assessment aimed at evaluating the opportunities for multi-RAT in Chapter 4.

Chapter 4

Providing Multi-RAT on Device

In Chapters 2 and 3, an examination was conducted on how LPWAN can be deemed suitable for establishing low-power, long-range connections. However, it should be noted that the properties of any individual LPWAN technology may not align perfectly with the use-cases proposed in Chapter 1.

Using multiple LPWANs in a multi-RAT solution can mitigate the limitations of any single LPWAN technology concerning the relevant use-case. Two approaches to achieving multi-RAT are studied: multi-RAT on the IoT node and providing multi-RAT across a network of IoT nodes. The former involves combining multiple LPWAN technologies on a single IoT node and is studied in this chapter. The latter focuses on establishing a network infrastructure that shares the usage of multiple LPWAN technologies, allowing devices to connect to the IoT by relaying information via other multi-RAT nodes and is studied in Chapter 5.

In this chapter, a custom build multi-RAT hardware platform that can serve a diverse set of use-cases is firstly presented. This platform can also serve for in-the-field parameter monitoring for LPWAN IoT networks. Next, opportunities are highlighted where multi-RAT support can benefit a variety of use-case requirements, with particular emphasis on energy consumption of the IoT node.

This chapter is based on the following publications:

- G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "Multi-RAT IoT – What's to Gain? An Energy-Monitoring Platform," In 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), Jun. 2023. [LCVD23]
- G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre and L. De Strycker, "Multi-RAT for IoT: The Potential in Combining LoRaWAN and NB-IoT," IEEE Communications Magazine, vol. 59, no. 12, p. 98-104, Oct. 2021. [LCO+21]

Contributions: For both papers, the author took the lead in conceptualizing, designing the appropriate hardware and firmware, conducting the experiments, and writing the papers. All co-authors provided assistance in writing the manuscripts.

4.1 Opportunities for Multi-RAT systems on IoT nodes

In its simplest form, IoT use-cases where slow-changing parameters need to be monitored can be adequately handled by a single RAT solution. Choosing any singular LPWAN connection is adequate in response to the – mostly static – requirements of the use-case (such as energy consumption, or QoS). However, as we delve into more intricate use cases with varying requirements over time (e.g., network coverage, or varying payload sizes), the limitations of a single RAT become evident.

These use-cases under varying operating conditions, may benefit from the integration of multiple wireless technologies. Integration of multiple wireless technologies on a single IoT node can lead to improved performances with respect to parameters such as energy consumption, latency, or throughput.

Such multi-RAT strategies have previously been used for short-range IoT devices, e.g., incorporating both Wi-Fi and Bluetooth connectivity [LKPC15]. Similarly, in cellular networks, different technologies are being used in a fall-back strategy. LPWAN technologies feature long-range connectivity whilst remaining relatively low power, as observed in Chapter 3. They vary, however, largely in terms of data bandwidth, QoS and energy-efficiency [LCC+17; MBCM19; ADA+19; dORRM19]. Unlicensed LPWAN technologies such as Sigfox and LoRaWAN have been combined into a single modem module. However, a true multi-RAT approach that allows for dynamic switching between LPWAN technologies, both unlicensed and cellular, is lacking.

By implementing a multi-RAT setup on an IoT node, its ability to communicate efficiently is significantly enhanced. This is primarily due to the node's capability to seamlessly switch between different connectivity options, taking into account various factors such as location, required throughput, or power consumption. This dynamic adaptation allows the node to optimize its energy expenditure:

- *Enhanced throughput:* Incorporating both high-throughput and low-throughput networks in a multi-RAT setup enables dynamic switching between the two networks, depending on the specific use-case. For instance, low-power, low-throughput networks can transmit keep-alive data, while large data frames can be transmitted over high-throughput networks.
- *Latency:* Time-series monitoring data, which can tolerate delays, can be transmitted through a high-latency RAT. On the other hand, threshold-based data, such as alarms that require quick delivery, can be transmitted through a low-latency RAT.
- *Reduced power consumption:* The IoT node can switch to a lower power consumption network when its battery is running low, allowing it to conserve energy and extend its battery life.
- *Enhanced security:* IoT devices can opt for different security protocols and encryption methods. For instance, cellular networks typically provide strong encryption and authentication mechanisms, while low-power wireless networks use more lightweight protocols.

Additionally, by leveraging this multi-RAT capability, the node can accommodate new application requirements:

- *Improved network coverage and availability:* The coverage provided by different IoTs RATs can vary significantly, and relying on a single technology may result in limited coverage and poor connectivity in certain areas. However, by integrating multiple IoT technologies into a single device, IoT nodes can effectively extend their coverage and improve their connectivity. Furthermore, in situations where IoT devices require global coverage, a multi-RAT setup becomes particularly crucial. In countries where there is no coverage of a specific technology or no roaming options available, the device can seamlessly switch to another RAT that provides coverage.
- *Improved reliability:* By leveraging multiple IoT technologies in a single device, IoT nodes can also benefit from improved reliability and redundancy. In the event of a network outage or other connectivity issues due to e.g., electromagnetic interference on a certain frequency band, the device can automatically switch to a another network.

– *Cost-effectiveness:* When it comes to IoT communication services, operational costs are typically calculated based on the number of messages sent or the amount of data transferred. To optimize costs, multi-RAT devices can switch between different networks based on their availability and cost. For instance, in track-and-trace use-cases, companies may have a local, private LPWAN network on-site, which can be used by the IoT node at almost no cost. When the node leaves the company site, it can switch to a public network and only then be charged for the incurring data transfers.

It is clear that, no matter how widely applicable any of the aforementioned LPWAN technologies may be, there is no one-fits-all technology. The remainder of this chapter assesses the benefits and drawbacks of a multi-RAT IoT solution. The presented multi-RAT platform, detailed in Section 4.2, enables measurement of particular use-case trade-offs and energy gains. An effective approach for providing multi-RAT is to combine the complementary IoT technologies NB-IoT and LoRaWAN on a single device. In contrast, Sigfox is considered less complementary due to its lower energy efficiency and specific feature set, in regard to LoRaWAN and NB-IoT. This is further elaborated in Section 4.3.

4.2 Platform for Dynamic Energy Consumption Measuring for Multi-RAT Setups

As discussed in Chapter 3, the energy consumption of an IoT node is determined by various parameters, both network and environment related. Literature focuses mainly on theoretical energy models or models obtained in lab conditions [MAC+20; SGIB20], omitting the non-negligible impact caused by vendor and operator-specific configurations or various coverage conditions [29; MABV19]. One publication by Michelinakis *et al.*, published in-the-field observations of large differences in energy consumption between various network providers of NB-IoT [MAC+20]. This highlights the need for platforms to obtain an accurate comparison and energy model of LPWAN IoT technologies, where both network and environment parameters are closely monitored.

4.2.1 Multi-RAT Platform

In order to quantitatively assess the potential of a multi-RAT solution, and investigate possible practical energy efficiency optimization approaches, a

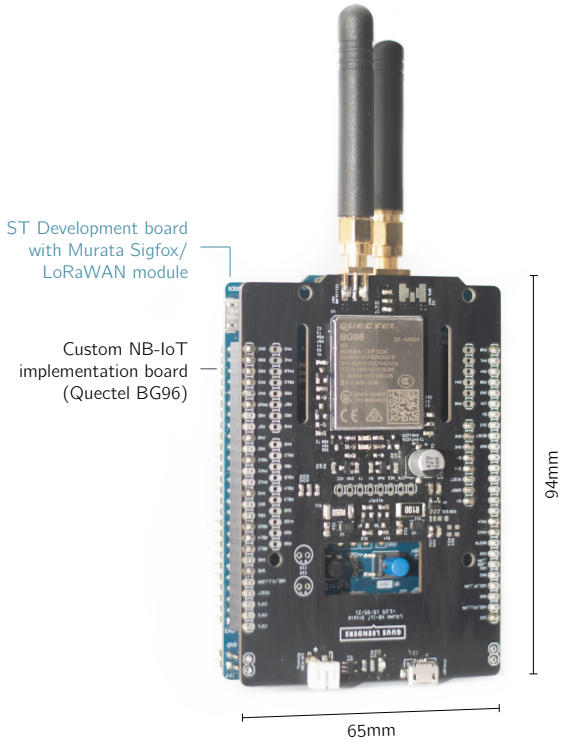


Figure 4.1: Picture of the custom and open-source multi-RAT platform [27] to perform per-packet energy measurements in NB-IoT, LoRaWAN, and Sigfox networks.

prototype that characterizes IoT wireless networks is created. The realized prototype includes LoRaWAN, Sigfox and NB-IoT modems and is able to measure energy consumption on-device on a per-packet-basis. These measurements are wirelessly transmitted and evaluated in a cloud-based back-end.

While other available hardware platforms with multi-RAT either lack a focus on low-power operation or do not incorporate the necessary LPWANs, there is one platform that bears comparison to our implemented hardware platform – Pycom FiPy. The Pycom FiPy offers a versatile combination of Sigfox, LoRa, Wi-Fi, Bluetooth, and NB-IoT. However, at the time of writing, the Pycom FiPy has become unavailable for purchase. What sets our developed prototype apart is its ability to control and monitor communication at a more granular level, particularly concerning LoRaWAN and NB-IoT. Additionally, we have incorporated in-the-field energy consumption measurements into our prototype. This renders our platform

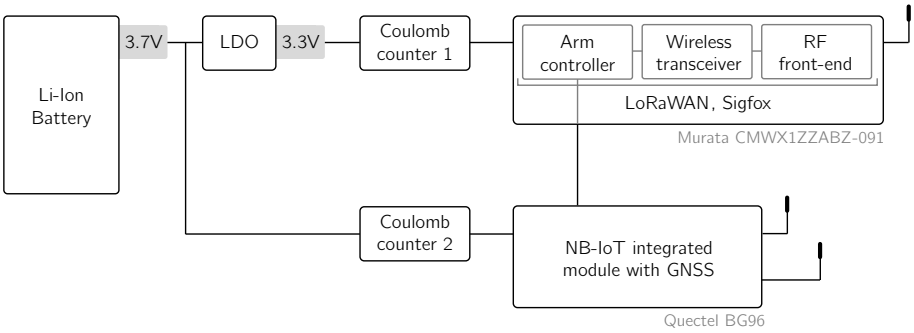


Figure 4.2: Schematic overview of the proposed system.

particularly compelling for researching the energy consumption of LPWANs.

Our developed prototype serves a dual purpose: firstly, it empowers IoT developers to tailor application requirements and restrictions by utilizing on-site energy consumption data. Secondly, it functions as an IoT device capable of dynamically switching between LPWANs based on changing circumstances dictated by the specific use-case, which is currently absent in existing literature. The prototype is depicted in Fig. 4.1.

On-Board Wireless Technologies. A schematic overview of the prototype testing platform is depicted in Fig. 4.2. The core of the device features the CMWX1ZZABZ-091 module by Murata. The module is powered by an ultra-low-power Arm Coretx-M0+ microcontroller (STM32L072CZ) and wireless transceiver (SX1276) [20] (LoRaWAN and Sigfox). The interface to program the microcontroller is user accessible, eliminating the need for an extra microcontroller [CLV+21]. The controller firmware was customized in such a way, that dynamic switching between LoRaWAN and Sigfox is possible. The CMWX1ZZABZ-091 module hardware is extended with NB-IoT hardware: a custom designed extension board featuring a Quectel BG96.

Power/energy measuring. Power distribution of the platform is illustrated in Fig. 4.2. To accurately measure energy consumption on a per-packed-basis, high accuracy coulomb counting is used. Two coulomb counting modules are used, measuring the power usage on each power rail. The coulomb counting algorithm is performed by the LTC2941 battery gas gauge and is set to the appropriate current resolution for each IoT technology.

Parameter Monitoring. The energy consumption is determined by the configuration parameters of each technology and the context (e.g., network coverage). In order to map the measured energy expenditure to these parameters, a detailed list of all actual applicable parameters is retrieved and transmitted to the back-end for further analysis.

Common parameters for LoRaWAN, Sigfox and NB-IoT are:

- number of received messages at the gateway/back-end,
- number of transmitted messages,
- payload size,
- time of reception,
- location of the receiving gateway(s)/base station(s),
- RF properties: RSSI, SNR, transmit power,
- GNSS position,
- motion speed.

For a LoRaWAN transmission, the current SF is recorded. The evolution of both the transmit power and the SF indicates the available LoRaWAN coverage when ADR is enabled. For Sigfox transmissions, the estimated region where the device is located, reported in the back-end, is collected. For all NB-IoT transmissions, the used CE level is stored, summarizing the network coverage. Other transmission specific data is also stored: RSRP, signal-to-interference-plus-noise ratio (SINR), and reference signal received quality (RSRQ). eDRX and PSM settings are recorded, to estimate the current power profile.

Monitoring Algorithm. The aforementioned on-board parameters and energy consumption per technology are acquired according to the approach depicted in Fig. 4.3. The measurement cycle consists of two parts, where first we transmit one packet per IoT technology, followed by a message reporting the captured results, including the measured energy consumption per packet. In the first phase, both the payload size and transmit power are varied per measurement cycle. The used transmit power depends on ADR in LoRaWAN and the network in NB-IoT. For Sigfox, the transmit power is fixed to 14 dBm. In every measurement cycle, a random payload size is selected per technology. The maximum payload size is different for each technology, i.e., 12, 256 and 1547 B for Sigfox, LoRaWAN and NB-IoT, respectively.

Power consumption is monitored from the start until the end of the wireless transmission. In NB-IoT especially, this includes eDRX or delayed RRC releases. Each of these packets gets sent to a custom cloud platform, either directly (NB-IoT) or through network operated cloud platforms (LoRaWAN and Sigfox). Power

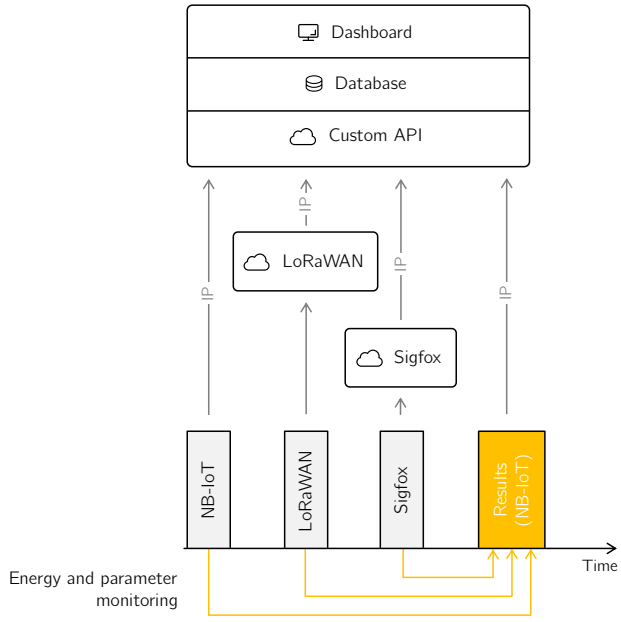


Figure 4.3: Monitoring algorithm overview.

consumption and various other previously discussed transmission parameters are recorded and sent to the custom cloud platform using an extra NB-IoT packet. This transmission is not included in the energy consumption metrics and is only used to communicate the stored measurements and configurations.

Dashboard. To ease the evaluation of the experimental data, a web interface for accessing real-time results was developed (Figure 4.4). Any relation between the discussed parameters (including consumed energy) can be analyzed by dynamically generating various graphs (e.g., scatter plots or maps). By applying multiple filters, IoT developers and researchers can easily focus on particular use cases to derive conclusions about how specific parameters impact energy consumption.

4.2.2 Experimental Evaluation

The proposed platform is experimentally validated by mapping the characteristics of several IoT networks in various circumstances (location, moving speed, indoor/outdoor and urban/rural). These circumstances correspond to a large

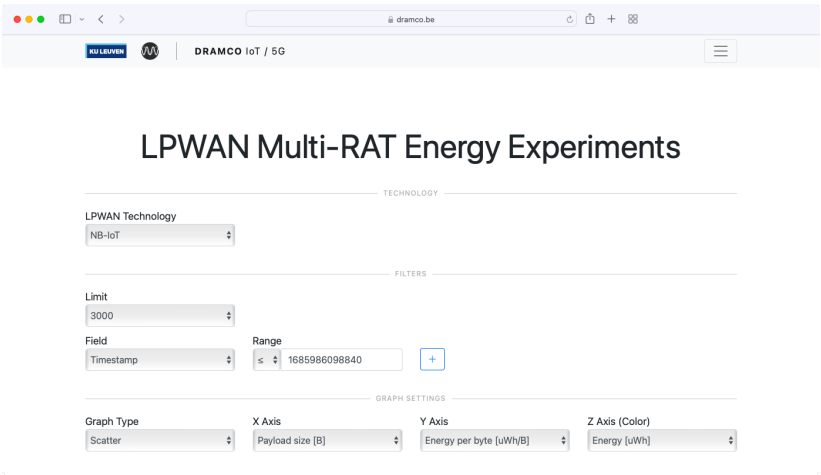


Figure 4.4: Screenshot of the dashboard developed to easily monitor the performance of the multi-RAT stup.

number of use cases: performing sensor ratings in smart cities at a fixed location, tracking the movement of (rental) bikes in cities or monitoring environmental tracking of sensitive assets during transport.

In this validation, we focus on two parameters: packet delivery ratio (PDR) and energy consumption per (payload) byte. The PDR and energy consumption are measured for every individual IoT technology. In the experiments, any confirmation of reception is disabled on all IoT technologies. Downlink parameters are enabled, though, only to enable energy optimizing strategies such as ADR in LoRaWAN. These experimental results were collected at various locations across the Belgian region of Flanders. In total, 5599 data points are gathered.The networks used are the Proximus NB-IoT and LoRaWAN network. CityMesh is the official Sigfox Operator in this region. All of these operators claim full coverage across the region. Static indoor and outdoor scenarios were captured at four discrete sites spanning the West- and East-Flanders regions, encompassing two urban and two rural locations. The mobile measurements were collected while commuting between these designated sites. Indoor scenarios are measured above ground level.

The results of the conducted experimental campaign are summarized in Figure 4.5 and present a comprehensive analysis of both PDR and energy consumption per byte across multiple scenarios and payload sizes. Figure 4.5 focuses on location scenarios, taking into account whether the nodes are stationary or mobile, and whether the environment is indoor or outdoor. These measurements encapsulate

the entire dataset, encompassing various geographical locations, rural and urban. Within this table, the indicated nominal value signifies the average of sample data for a specific configuration. It's worth noting that the number of measurements can fluctuate as the total number of measurements are distributed across distinct payload size groups: e.g., NB-IoT segments its payload into four distinct ranges, whereas all Sigfox data points are consolidated into a single data point. To provide a statistical evaluation of these measurements, the quartiles are also included in the table, in addition to the sample count for each setting. The complementary cumulative distribution function (CCDF) of the measurements for payload sizes of 1 B to 12 B are depicted in Figure 4.6 and Figure 4.7. In order to assess the contrast between urban and rural scenarios, a comparison of the PDR in the static outdoor setting is made for both urban and rural environments, as detailed in Table 4.1. This table encompasses the complete dataset for the specific scenario in question, irrespective of the transmitted payload size. It's important to highlight that, given the restricted number of locations available for static scenarios, any conclusions drawn regarding metrics between indoor and outdoor scenarios may carry lower statistical validity. The primary purpose of these findings is to facilitate a comparative assessment of the performance of the three LPWAN technologies in these specific scenarios.

Based on the gathered data, it is clear that NB-IoT outperforms both LoRaWAN and Sigfox in terms of QoS, as indicated by consistently higher PDR values for NB-IoT. This advantage can be attributed to the cellular nature of NB-IoT, which provides a dual benefit. First, it enables bidirectional signaling between the UE and the base station, leading to improved QoS. Second, NB-IoT leverages a well-established network infrastructure that offers broader coverage. Following the RA procedure in NB-IoT, a series of messages are exchanged to configure the connection, notably including the system information block (SIB) messages. Through these communication messages, the connection parameters are adapted based on measured channel properties, such as the RSRP. Decisions are made regarding parameters like CE level and the number of repetitions, among others. This approach differs from the methods employed by LoRaWAN and Sigfox. These technologies use ALOHA-based protocols, where messages are typically dispatched without synchronization or extensive pre-configuration by the base station prior to data transmission. [1] It's worth noting that enabling acknowledgements of uplink messages in LoRaWAN and Sigfox could provide some improvement in QoS, yet this is only after actual data transmission occurs, and was not the case in the measurements conducted here.

[1]The ADR algorithm in LoRaWAN provides a notable exception, yet it requires multiple data exchanges and does not guarantee a level of QoS before actual data transmission occurs.

Figure 4.5: Comparison between NB-IoT, LoRaWAN and Sigfox as experimental validation of the presented multi-RAT platform. Packet delivery ratio (PDR) and the average energy per byte E_b serve as comparison metrics. Cells indicated with '-' signify that the payload size is unavailable in the respective network, while cells indicated with '/' suggest that this particular combination has not been observed in the measurements conducted.

Payload Size (B)		Static						Mobile		
		Indoor			Outdoor			Outdoor		
		NB-IoT	LoRaWAN	Sigfox	NB-IoT	LoRaWAN	Sigfox	NB-IoT	LoRaWAN	Sigfox
1-12	PDR (%)	92.78	62.16	70.76	90.91	52.89	83.00	92.15	55.35	38.99
	Q1 | Q2 | Q3	100 | 100 | 100	27 | 75 | 100	70 | 87 | 100	100 | 100 | 100	0 | 53 | 83	64 | 100 | 100	100 | 100 | 100	33 | 66 | 100	16 | 33 | 55
	E_b (µW h/B)	41.10	8.03	51.89	50.19	11.64	54.78	63.46	15.25	66.05
	Q1 | Q2 | Q3	35 | 37 | 46	3.8 | 7 | 11	42 | 50 | 65	37 | 39 | 50	9 | 11 | 14	44 | 53 | 64	46 | 56 | 86	9 | 12 | 18	56 | 59 | 72
	# Packets	127	483	998	54	882	1000	33	102	217
12-51	PDR (%)	90.58	65.47	-	85.65	52.60	-	82.98	57.50	-
	Q1 | Q2 | Q3	85 | 100 | 100	0 | 63 | 100		78 | 100 | 100	0 | 63 | 100		63 | 100 | 100	0 | 50 | 83	
	E_b (µW h/B)	23.94	3.64	-	18.74	6.15	-	29.68	5.85	-
	Q1 | Q2 | Q3	18 | 23 | 27	2.1 | 3.6 | 4.5		12 | 16 | 24	5.1 | 6.2 | 7		20 | 28 | 39	3.7 | 5.3 | 8.2	
	# Packets	445	227		170	172		46	39	
51-255[a]	PDR (%)	98.55	69.57	-	89.02	/	-	94.44	/	-
	Q1 | Q2 | Q3	100 | 100 | 100	58 | 63 | 81		83 | 100 | 100			100 | 100 | 100		
	E_b (µW h/B)	5.98	0.33	-	4.12	/	-	10.12	/	-
	Q1 | Q2 | Q3	4.1 | 5.7 | 8.1	.17 | .23 | .39		2.8 | 3.7 | 5.4			4.8 | 7 | 10		
	# Packets	22	68		163			17		
255-1547	PDR (%)	100	-	-	92.29	-	-	83.33	-	-
	Q1 | Q2 | Q3	100 | 100 | 100			100 | 100 | 100			40 | 90 | 100		
	E_b (µW h/B)	1.03	-	-	0.81	-	-	0.89	-	-
	Q1 | Q2 | Q3	.56 | .89 | 1.4			.39 | .73 | 1.1			.52 | .73 | .96		
	# Packets	79			194			11		

[a] For LoRaWAN, this range is only available under favorable network conditions, i.e., a LoRaWAN SF 9 or lower.

Table 4.1: Comparative analysis of the PDR evaluation metric for NB-IoT, LoRaWAN, and Sigfox. This data is a detailed excerpt from Figure 4.5. This analysis focuses on static outdoor scenarios, here with a distinction between urban and rural environments, regardless of payload size.

	Urban			Rural		
	NB-IoT	LoRaWAN	Sigfox	NB-IoT	LoRaWAN	Sigfox
PDR	93,42	73,29	85,61	94,08	41,89	81,13
Q1 | Q2 | Q3	90 | 94 | 100	60 | 67 | 80	70 | 77 | 100	90 | 91 | 100	0 | 37 | 64	55 | 78 | 100
# Packets	304	483	528	304	663	673

Based on this data, four opportunities for multi-RAT enabled communication can be observed:

1. *Payload dependent energy consumption.* Energy consumption per byte (E_b) for smaller messages (1 B - 51 B), improves slightly when comparing Sigfox to NB-IoT but at least quadruples when comparing LoRaWAN to NB-IoT. This makes LoRaWAN an attractive choice for transmitting small data packets efficiently. To handle larger packets (51 B - 1547 B) without the need to fragment payloads across multiple LoRaWAN packets, the addition of NB-IoT alongside LoRaWAN on an IoT device is suggested. This setup also alleviates potential latency concerns that may arise when data is divided and sent in multiple LoRaWAN messages. Thus, IoT applications with varying payload sizes can significantly benefit from implementing a multi-RAT scheme (combining NB-IoT and LoRaWAN), which offers greater energy efficiency across diverse use cases.

2. *Coverage service area.* It is evident from the data presented in Table 4.1 that both LoRaWAN and Sigfox exhibit a decline in PDR when comparing urban and rural scenarios. In cases where LoRaWAN – which is preferred due to its superior energy efficiency for sending small messages – coverage might be insufficient, such as in rural areas, nodes could consider switching to NB-IoT to ensure connectivity, although at a lower energy efficiency.

3. *Mobility.* IoT communication on mobile nodes can suffer from low PDR when moving at high speed. By comparing static to mobile measurements, it is clear that the PDR of Sigfox dramatically diminishes for mobile nodes. This can also be seen when plotting the PDR versus speed in the measurement platform dashboard (Fig. 4.8). For Sigfox, PDR drops with increasing speed, while the PDR of both NB-IoT and LoRaWAN largely remains constant. These results are consistent with measurements gathered by S.-Y. Wang *et al.* [WCFS20].

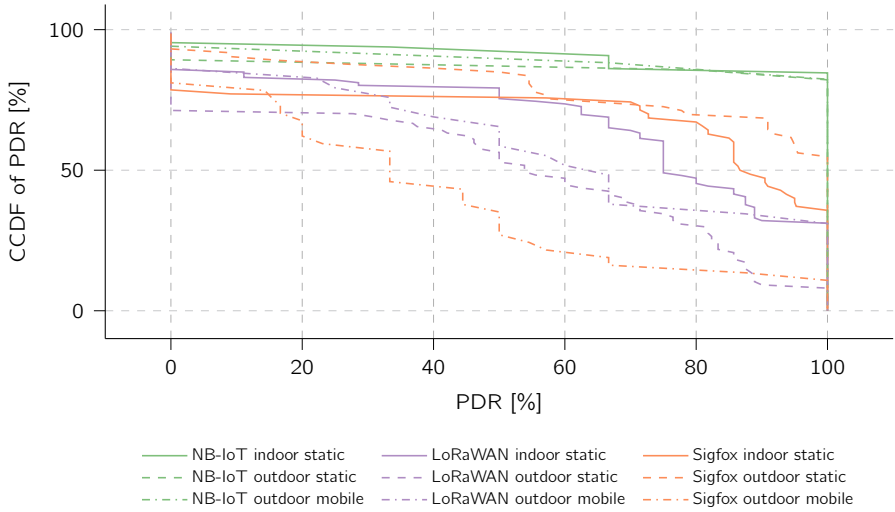


Figure 4.6: CCDF of PDR for the different scenarios illustrated in Figure 4.5, covering payload sizes from 1 to 12 bytes. This graph illustrates the distribution of PDR measurements, which are grouped in series of packets sent within a 1-hour timeframe.

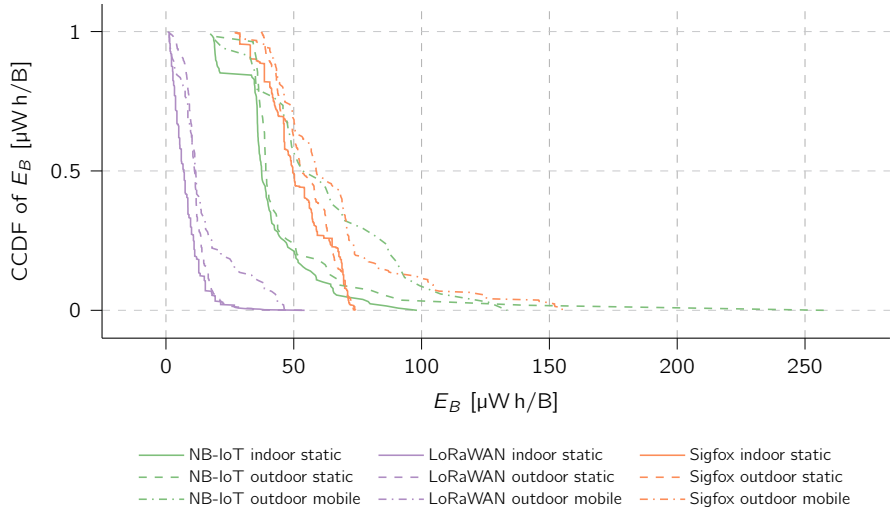


Figure 4.7: CCDF of the energy per byte E_b for the different scenarios illustrated in Figure 4.5, covering payload sizes from 1 to 12 bytes.

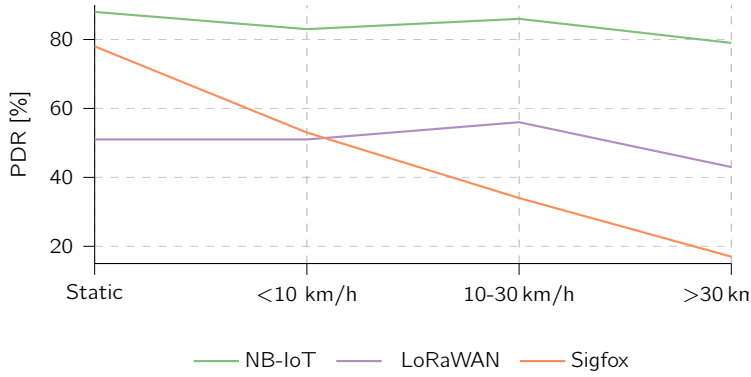


Figure 4.8: Comparison of packet delivery ratio (PDR) of NB-IoT, LoRaWAN and Sigfox in function of speed of the IoT node for packet payloads of 1 B-12 B.

4. *Mission-critical IoT.* Guaranteeing delivery in IoT requires a downlink channel for confirmation packets to be sent. Traditionally, one would opt for a high PDR IoT technology such as NB-IoT. However, the energy cost of sending a NB-IoT packet is relatively high when compared to LoRaWAN. First, attempting to send critical data via LoRaWAN (with a confirmation downlink channel) can improve energy consumption drastically.[2]

4.2.3 Conclusion: Combining IoT technologies

When comparing LoRaWAN and Sigfox, the energy consumed by LoRaWAN is found to be at least four times lower than that of Sigfox for equal payload sizes. This significant difference in energy consumption makes LoRaWAN an attractive choice for energy-efficient applications. In terms of Packet Delivery Ratio (PDR), Sigfox outperforms LoRaWAN for static use-cases. This indicates that Sigfox provides considerably better network coverage, making it a suitable option for applications where a more reliable message delivery is wanted. On the other hand, NB-IoT demonstrates better PDR than Sigfox, whether static or mobile, while achieving a comparable energy consumption. Taking these factors into account, in the context of PhD, Sigfox is considered not complimentary to other LPWANs and thus less suitable to incorporate into a multi-RAT setup.

[2]Here, it is assumed that a latency of up to 10 s is tolerable. When combining this approach with periodic monitoring, it is crucial to consider the duty-cycle limits imposed to ensure that additional messages can still be sent in the event of alarms.

In the context of this PhD, the two LPWAN technologies that are deemed most complementary and are therefore best suited for integration into a multi-RAT setup for remote IoT nodes are LoRaWAN and NB-IoT. Switching from NB-IoT to LoRaWAN can save considerable amounts of energy for non-critical, small messages. On the other hand, NB-IoT outperforms LoRaWAN with respect to coverage and data throughput. In this regard, Section 4.3 delves into the exploration of an effective approach to achieve a successful combination of two complementary RATs, LoRaWAN and NB-IoT, in order to establish a multi-RAT setup.

4.3 The Potential in Combining LoRaWAN and NB-IoT

In this section, the potential of combining LoRaWAN and NB-IoT in a multi-RAT solution for IoT is analyzed. To that end, we evaluate key IoT node requirements beyond only energy efficiency in function of payload size and link quality: (i) energy efficiency, (ii) coverage, (iii) payload size, (iv) latency performance, (v) quality of service, and (vi) cost efficiency.

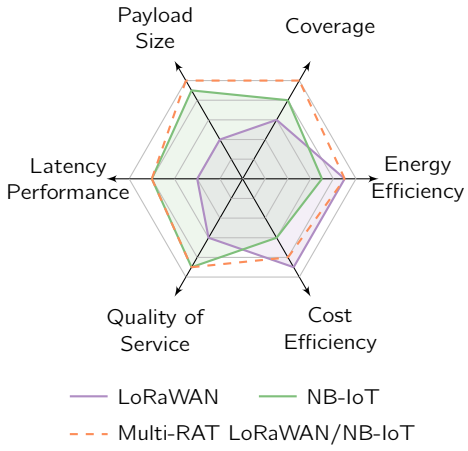


Figure 4.9: Comparative study of the explored IoT network technologies, presenting the main IoT feature requirements.

4.3.1 LPWAN Multi-RAT support for IoT Requirements

The IoT specifications, presented in Figure 4.9, are used to analyze the main properties of LoRaWAN and NB-IoT, thereby assessing the potential multi-RAT opportunities. Below, these characteristics are elaborated upon for each technology impacting the considered application requirements.

4.3.1.1 Energy Efficiency

Both LoRaWAN and NB-IoT provide measures to minimize energy in support of battery-powered devices, as described in Chapter 2 and Chapter 3. A multi-RAT system for IoT should select the most energy-efficient IoT technology within a set of application parameters (e.g., payload, latency tolerance).

LoRaWAN. LoRaWAN is low-power by design. The simple MAC and narrow bandwidth signals contribute to a reduced power consumption. As concluded in Section 3.1, when utilizing LoRaWAN, the energy consumption is mainly determined by the airtime which depends on (i) the payload size and (ii) the spreading factor. The adaptive data rate algorithm implemented in LoRaWAN can improve energy consumption of the IoT node. This algorithm alters the transmit power and data rate of the end-devices depending on the wireless channel conditions.

NB-IoT. To address typical LPWAN-specific requirements, the LTE standard was simplified with an emphasis on energy reduction and lower complexity. Most sleep timers have been extended in the release of NB-IoT in comparison with LTE. In this manner, the end-devices can sleep for a longer time period before making contact to the network, as explained in Section 3.3. The major energy improvements are due to the introduction of (i) eDRX which reduces the amount of paging, (ii) PSM, allowing to disconnect from the network while still maintaining context for fast re-connections.

4.3.1.2 Coverage

When deploying IoT nodes in a city or in remote locations, good coverage is critical for obtaining a reliable communication. Both LoRaWAN and NB-IoT feature mechanisms to improve range and coverage. The distance to the nearest gateway

clearly is the main determining factor for reliability for both of the technologies. The networks are typically not deployed from a common infrastructure. Hence, there is inherent redundancy in a multi-RAT solution and the reliability can be greatly improved with respect to any single RAT IoT connectivity.

LoRaWAN. The range strongly depends on the utilized data rate. Increasing the spreading factor lowers the required demodulation floor and thereby extends the range. The LoRa link budget can be as high as 156 dB [WKL+16] (SF 12). In contrast to cellular networks, private networks and crowd-sourced networks can be used in LoRaWAN. The coverage can simply be extended by adding more gateways to the network.

NB-IoT. Three CE levels implemented in NB-IoT. They enable providers to provide network connectivity in places that are hard to reach, such as subterranean parking lots. This way, the MCL is improved by 20 dB, compared to GPRS or LTE: up to 164 dB. The observed gain is primarily due to the substantial increase in repetitions under adverse signal conditions, reaching up to 128 repetitions compared to only 2 repetitions under good conditions [8]. Uplink messages are typically transmitted at 23 dBm [AAP+17].

4.3.1.3 Payload Size

Implementing a multi-RAT solution provides the combined range of available payload sizes.

LoRaWAN. The payload size defined in LoRaWAN depends on the utilized data rate. For higher spreading factors, LoRaWAN restricts the payload size to 51 bytes, while in more optimal conditions a 242 B payload can be used in one message (both in uplink and downlink). In case a longer payload is required, the developer has to divide the payload over multiple messages, thereby increasing the reception latency and energy consumption.

NB-IoT. The maximum payload size for each message, regardless of conditions, is 1600 B [MBCM19], for both uplink and downlink messages. There are no duty cycle limits imposed in NB-IoT: an unlimited amount of messages can be sent per day. However, the amount of data you are allowed to consume is subjected to the data plan of the provider.

4.3.1.4 Latency

We consider two latency requirements, i.e., uplink and downlink latency. Uplink latency is here defined as the time between the intent of sending a message on the device and receiving that message on the server (back-end). This latency, in practice, can be largely attributed to the latency between the IoT node and the gateway, which in practice is dominantly determined by the node-gateway link. Notably, there is a considerable difference between the intent of sending a message and actual sending the message in license-exempt bands due to duty cycle limitations. Downlink latency is defined as latency in the reverse direction.

LoRaWAN. The latency of the downlink messages is constrained by the adopted device class, while the latency of the uplink messages is limited by the duty cycle and data rate.

- Due to license-exempt operation, the devices are subjected to a maximum duty cycle[3]. This increases the latency of uplink messages as they have to respect a minimum wait time before transmitting the next message.

  Furthermore, the data rate has an impact on the uplink latency. The data rate of LoRaWAN depends on the utilized SF, bandwidth and coding rate (CR). The time on air of a LoRaWAN package, ranges from 25 ms (SF7, 1 B) to 2.5 s (SF12, 51 B). To put this in perspective, when having a duty cycle of 1%, we need to wait 99 times the airtime. When we send a 1 B packet with SF7, this means that we have a minimum uplink latency of almost 2.5 s. In case of a 51 B message at SF12, this becomes 4 min 8 s.[4]
- Downlink latency in LoRaWAN varies based on device class: A, B, or C. Class A requires sending an uplink message before receiving downlink messages. Class B has scheduled windows for downlink reception. Class C has continuous downlink listening for minimal latency. Consequently, applications requiring low latency downlink messages should adopt class C. Class B devices should be used when there are more relaxed constraints on the latency, but more downlink messages are still required.

[3]This is commonly 1% for EU 868 MHz and 30 seconds per day imposed by the fair access policy of the The Things Networks.

[4]Here we consider that there is only one 1% duty cycle channel available.

NB-IoT. While uplink latency is limited to 10 s at most [RMZ+16] in NB-IoT[5], downlink latency heavily depends on the timer values used by eDRX and PSM (Table 3.6).

- Uplink latency is mainly influenced by the path loss and the deployment method of the base station [LSW+17]. An extensive NB-IoT latency model has been documented by Azari *et al.* [AMSP18]. When a device needs to send an uplink message, it first needs to listen for cell information. Through this information gathering, the node synchronizes with the base station. By sending a random access request to the base station, the device performs access reservation. The base station responds by sending a random access response, indicating resources reserved for the NB-IoT transfer. Finally, the device is able to send data to the base station.
- Both eDRX timers and PSM timers regulate when the node is able to receive data, thus controlling downlink latency. By prolonging the eDRX cycle, more periodic paging cycles occur. When in PSM, no packets can be received until the TAU message is sent.

Typical latency figures reported in literature [LSW+17] range from 0.3 s to 8.3 s, depending on link budget and deployment type. In a stand-alone deployment scenario, full base station power is available to NB-IoT, improving latency. In good coverage conditions, latency is predominantly caused by the time to acquire synchronization and waiting for an access opportunity. Latency in poor coverage conditions is generally caused by latency of the exception report.

4.3.1.5 Quality of Service

In typical IoT use cases, QoS focuses on packet loss and throughput. Mechanisms such as enabling repetitions, increasing output power, etc. allow technologies to control QoS dynamically.

LoRaWAN. The increased interference due to license-exempt operation, is addressed by employing the spread spectrum technique CSS. LoRaWAN does not support different QoS levels. However, an acknowledgment can be requested so the device can retransmit a message if required.

[5] Notably, this is specified for exception reports which are high-priority packets made specifically for sending alerts.

NB-IoT. As NB-IoT operates in licensed spectrum, it can offer greater QoS than networks operating in an unlicensed spectrum. Furthermore, NB-IoT employs the same proven time slotted synchronous protocol as used in LTE ensuring end-to-end QoS.

4.3.1.6 Cost Efficiency

Evidently, the initial hardware investment cost of a multi-RAT solution is higher than for a single radio. Several costs need to be considered to get the total cost of ownership: spectrum cost, network deployment cost, the cost of the end device, and eventual costs related to replacement or recharging of batteries involving service visits.

LoRaWAN. Because LoRaWAN operates in the unlicensed bands, there are no costs related to the spectrum usage. However, and similar to NB-IoT, operators have invested in deploying a network, accessible if you have a subscription plan. Besides using others infrastructure, private networks can be deployed.

NB-IoT. Both the spectrum license and base station contribute significantly to the combined cost of a NB-IoT network, due to NB-IoT running in licensed spectrum. To illustrate, the cost of acquiring the necessary spectrum can amount to 500M€ per MHz [MBCM19].

4.3.2 Experimental Evaluation

LoRaWAN and NB-IoT have been thoroughly compared and documented in literature; however, experiments-based comparisons of both *energy consumption* and *latency* are lacking. In what follows, these parameters are assessed in the field.

4.3.2.1 Experiment Setup

The performed LoRaWAN experiments were conducted with the hardware presented in Section 3.1.1. The NB-IoT measurements were performed using a custom-made NB-IoT module, presented in Section 3.3.3.1. In our setup, it is possible to associate a specific modem state to the energy profile. By doing so, a

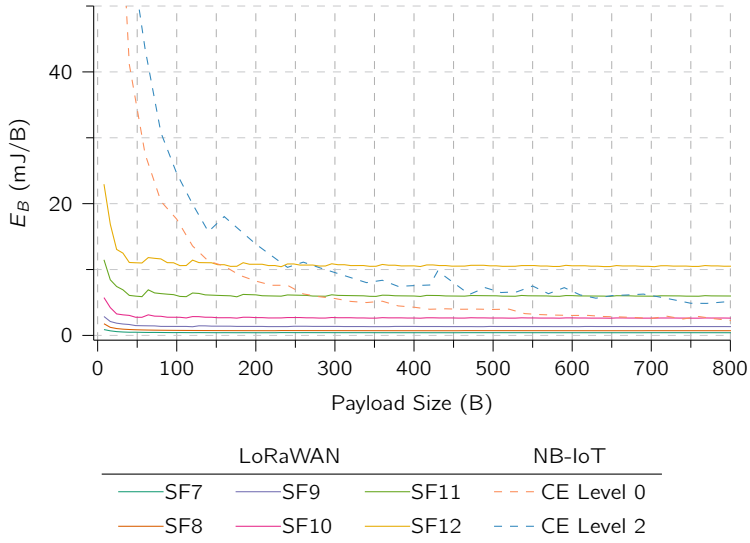


Figure 4.10: Energy consumption per byte comparison between LoRaWAN and NB-IoT. The difference in energy consumption is most noticeable when sending small payloads. At 150 B, NB-IoT is more energy efficient per byte with respect to LoRaWAN SF 12.

thorough profile of the NB-IoT energy consumption has been obtained. All files are open source [28] to facilitate reproducibility and adaptability.

4.3.2.2 Energy Efficiency

In this evaluation, the energy measurements for the power profiles of both LoRaWAN and NB-IoT are derived from the data obtained in Chapter 3. In order to compare the energy efficiency of LoRaWAN and NB-IoT, the energy per payload byte E_B has been assessed. Energy measurements only consider the energy used by the IoT transceiver and do not include the energy footprint of the peripheral hardware.

LoRaWAN. As observed in Chapter 3, the main energy expenditure constitutes the transmission of a packet, which is followed by two receive windows in LoRaWAN. Figure 4.10 shows the energy per payload byte for different payloads sizes.

NB-IoT. NB-IoT supports a maximum of 1600 B to be sent in a single packet. The effective payload size also influences the energy efficiency of NB-IoT. As fixed energy costs, such as cDRX and eDRX, are equal for smaller and larger payload sizes, the energy per byte (E_B) is lower for longer packets (Fig. 4.10). Notably, NB-IoT (at CE level 2) is more energy efficient than LoRaWAN (at SF 12) when the payload size exceeds 240 B. For payloads larger than 240 B, NB-IoT becomes progressively and up to 7 times more energy efficient than LoRaWAN with larger payloads. The energy impact of rising CE levels, however, is reduced when transmitting large payloads.

4.3.2.3 Latency

LoRaWAN. The latency with respect to the data block size is depicted in Figure 4.11. This calculation depicts the case when only the default and mandatory bands are used, i.e., 1% duty cycle. The uplink latency is limited by the allowed air time or, equivalently, duty cycle. The uplink latency for a payload size of 1600 bytes ranges from 1.2 minutes to 47 minutes (data transmission is distributed across the default and mandatory bands, see Figure 4.11). As the message size is constrained by the spreading factor, the payload needs to be split up in multiple messages in order to send the full payload, increasing the uplink latency. The total duty cycle is limited by the supported operating bands of the network.

NB-IoT. According to NB-IoT specifications, the latency should be kept under 10 s. This was experimentally validated by sending 1236 packages in varying signal conditions. The results, depicted in Fig. 4.12, show that both CE levels 0 and 1 keep latency within the 10 s maximum. Packets transmitted in CE level 1 only have a slightly larger latency than packets transmitted in CE level 0. When transmitting on CE level 2, however, poor signal conditions can cause latency to reach up to 20 s. The median latency increases and more extreme latency outliers occur. We also observed that the latency in NB-IoT is barely influenced by payload size.

4.3.3 Potential Multi-RAT Gains

Considering the predominant requirements for an LPWAN node, our assessment has highlighted the complementarities and potential synergies of LoRaWAN and NB-IoT technologies. Combining LoRaWAN and NB-IoT can greatly improve a node's functionality and energy efficiency. Based on our study and experimental

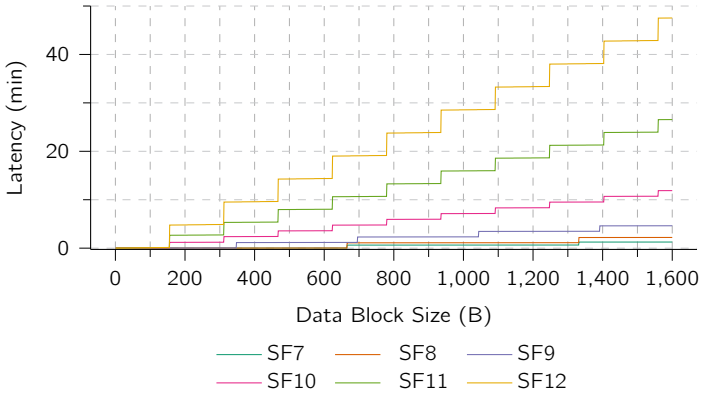


Figure 4.11: Latency per data payload size (multiple payload frames transmitted sequentially) for LoRaWAN. Data transmission is distributed across the default and mandatory LoRaWAN bands, adhering to a 1% or 0.1% duty cycle per band as imposed by the ISM regulations. Note that the latency can increase when network operators impose extra data limiting measures (e.g., fair access policy for The Things Network).

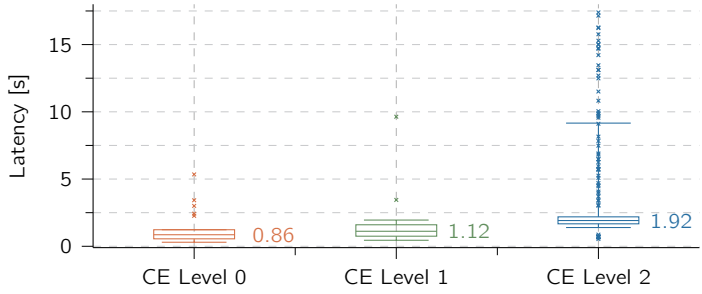


Figure 4.12: Boxplot of measured NB-IoT latency depending on the CE level when sending a packet: 14 B payload, confidence interval 95% is marked.

validation, we demonstrate the potential of a multi-RAT (LoRaWAN/NB-IoT) solution from the perspective of different IoT requirements, while focusing on an optimal energy trade-off.

Energy efficient operation for variable payload sizes. IoT use cases with varying payload sizes benefit from a multi-RAT approach. By implementing NB-IoT, messages up to 1600 B can be sent, while still enabling extremely low-energy messages with small payloads over LoRaWAN.

Use-case application – Smart city sound surveillance

In the context of smart city surveillance, consider the case where a node would send hourly sensor data, e.g., sound level, and more extensive data when certain thresholds are met such as loud noises. The multi-RAT solution would select LoRaWAN for hourly sensor updates (16 B, 97% of packages), and NB-IoT for sending more elaborate sensor recordings for classification (assumed 8 kB, spanning a total of 6 NB-IoT messages and occurring for example 5 times per week or 3% of sent messages). This multi-RAT solution would result in 79.8 J transmit energy per week, a reduction by a factor of 5 with respect to only using NB-IoT and of a reduction of 33 % when only using LoRaWAN. The latter would require the large payloads being split into 160 messages, which would also introduce a latency penalty of at least 1 hour when considering ETSI regulations and the used SF to be 10. To make this more tangible, this would mean that the device could, in ideal conditions, operate for 4.8 years on two AA batteries (for a total of 2500 mAh) when employing multi-RAT technology, 1 year when only transmitting via NB-IoT, and 3.4 years when only employing LoRaWAN.[a] This use-case is implemented in practice in Chapter 6.

[a]This energy calculation assumes a LoRaWAN SF of 10, the NB-IoT CE level to be 0, and includes the self-discharge energy of the battery as well as the sleep current, yet does not incorporate the current consumption of the sound sensor.

Guaranteeing timely delivery for latency-critical messages. Depending on network coverage and network load, latency can be optimized by spreading communications over multiple IoT technologies. When low energy consumption is critical and the NB-IoT chipset is in PSM, a faster wake-up can be achieved with LoRaWAN. On the other hand, larger payloads can be sent more rapidly with NB-IoT. This could be important for monitoring medical grade parameters (e.g., heart rhythm or fall detection).

Redundant Networking and Improving Service Area By combining multiple IoT technologies, the effective service area of a multi-RAT solution is extended to the area of all IoT technologies on-board. When coverage is not provided by one IoT technology, another can step in. The service area can be privately extended for LoRaWAN by deploying private gateways.

Improving quality of service. By enabling an IoT device to operate on both LoRaWAN and NB-IoT, the optimal QoS for any message can be chosen: increasing robustness and reliability. Periodic 'alive' messages do not need high QoS, yet more important messages containing sensitive data (e.g., temperature tracking on track and trace applications) do need high QoS. LoRaWAN should be used for the periodic, low QoS, messages, saving energy. Important messages can be sent through the NB-IoT network, featuring high QoS.

4.3.4 Multi-RAT Drawbacks

Device Footprint By including multiple IoT technologies on one device, multiple modems need to be on-board. This increases the space needed for wireless interfaces by both the applicable modem size and the appropriate antenna size. The board space occupied by multiple modems, can be improved by efficiently using both sides of a PCB. Multiple antennas can be combined in multi-band antennas.

Device Cost As multiple modems are included, hardware and network subscription costs rises. However, by optimizing energy consumption, costs can be saved by not requiring manual intervention for battery replacements.

Computational Overhead To dynamically switch between IoT technologies, some computational overhead is required. The energy savings from implementing a multi-RAT platform. However, the energy savings of multi-RAT outweighs the energy consumption of the required additional computations [KW07].

4.4 Conclusions

This chapter established the opportunities and challenges in combining multiple LPWANs in a multi-RAT solution. A custom multi-RAT platform has been created which includes dynamic energy consumption measuring capabilities for LoRaWAN, Sigfox and NB-IoT. An in-depth comparison is made in regard to energy consumption of each technology in combination with other network-related parameters such as network PDR.

Results show that combining LoRaWAN and NB-IoT in a multi-RAT solution, would proof the most beneficial. The potential in combining LoRaWAN and NB-IoT have been explored.

Energy savings can particularly be obtained when switching between LoRaWAN and NB-IoT because of the payload size that needs to be transmitted or the need for particularly high QoS. If large payload sizes only need to be transmitted occasionally, energy savings can be obtained by combining the two LPWANs, while minimizing latency. Use-case related causes, such as improving the service area and coverage, can also merit the implementation of LoRaWAN and NB-IoT on a single IoT node.

The employment of multi-RAT also entails certain drawbacks, including device and subscription costs, which should be considered. To address this concern, in scenarios involving a large group of nodes, the multi-RAT connection of one device in the network can be shared with other devices, mitigating costs. This approach is discussed in detail in Chapter 5. The considerations in regard to how to employ LPWAN multi-RAT technologies on IoT nodes, are further elaborated upon when being applied to specific use-cases such as the tracking of rental bikes, or the monitoring of sound in smart cities. This is discussed in Chapter 6.

Chapter 5

Multi-RAT Supported IoT Networks

To combat the drawbacks that are related to implement a multi-RAT system (see Section 4.3.4) across a plethora of IoT nodes in a single network, multi-RAT connectivity can be shared by a single, central, device in the network: a gateway. However, traditional gateways consume too much energy to be sustainable on battery power. To combat this, a new network protocol is implemented, relying on a multi-hop LoRa network to get sensor data to a central, NB-IoT connected, node. In this chapter, the network is conceptualized, simulated, and experimentally validated based on reliability and energy consumption.

This chapter is based on the following publications:

- G. Leenders, G. Ottoy, G. Callebaut, L. Van der Perre and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling," IEEE Wireless Communications and Networking Conference, 2023, doi: 10.1109/WCNC55385.2023.10118770. [LOC+23]
- G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling for Remote Sensing," MDPI Sensors, vol. 23, no. 11, p. 4994, May 2023, doi: 10.3390/s23114994. [LCO+23]

Contributions: For both papers, the author took the lead in conceptualizing, designing the appropriate hardware and firmware, conducting the experiments, and writing the papers. G. Ottoy assisted in writing firmware, and G. Callebaut

assisted in developing the required simulator framework. All co-authors provided assistance in writing the manuscripts.

5.1 Sharing Multi-RAT Connections

Enabling multi-RAT support on IoT nodes can increase device size, weight, and costs due to the incorporation of multiple modems. Its implementation should be carefully considered for specific use cases where the benefits outweigh the additional expenses. In some scenarios, it may be more practical to provide multi-RAT support to only a few devices within a network. To utilize multi-RAT capabilities, other nodes within the network must relay their communication through a central node that has the necessary multi-RAT capabilities.

In its simplest form, this is achieved through a gateway supported network, which is most commonly used in LPWAN networks. A central gateway provides cloud connectivity to all surrounding nodes via one technology, such as LoRa, Bluetooth, or ZigBee. The cloud connectivity is then established using another (wireless) technology, such as Wi-Fi or LTE. This chapter focuses on these gateway nodes that operate within the IoT scope as outlined in Section 1.1.2. Particularly, these nodes are also energy constrained and deployed remotely, and thus operate on a strictly limited energy budget which imposes strict limitations on the network.

5.2 Multi-Hop Wireless Networks

A gateway in LPWANs is commonly connected to the cloud using a single-hop star network topology. Despite the 2–5 km range of coverage that LoRaWAN can accomplish [PMR+15b], it might not be sufficient to cover the hundreds of square kilometers required in remote scenarios, or when propagation is severely attenuated, e.g., because of obstructed paths. Coverage could be provided by distributing a large amount of LoRaWAN gateways in the area, however, this proves to be unfeasible due to lacking (cellular) network coverage, energy restrictions, or financial cost. Therefore, a LoRa[1] based multi-hop network to extend the coverage is proposed, specifically optimized for a low energy consumption.

[1] LoRa is selected for its exceptional range, a key characteristic of LPWAN technologies. A BLE mesh network could proof to be a viable alternative to LoRa in this implementation. However, opting for this technology, would typically reduce the permissible distance between nodes as the range of BLE is inherently shorter than LoRa.

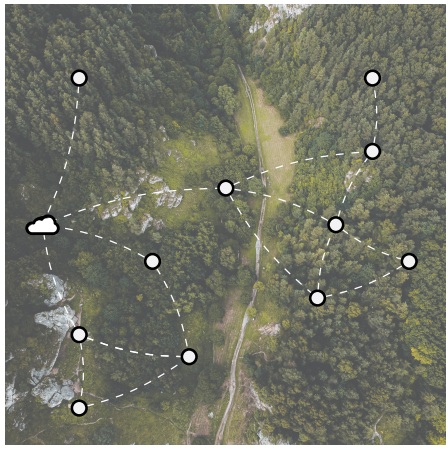


Figure 5.1: A multi-hop network deployment in a remote environment, designed for environmental tracking purposes in forests or nature reserves.

A multi-hop protocol allows messages to be transmitted from one node to another through a series of intermediate nodes, effectively allowing packets to *hop* from one node to the next until they reach their destination (see Figure 5.1). This enables the network to cover a much larger area than would be possible with a single hop. In this implementation, we assume only one possible destination: the gateway or the sink, which can store the gathered data or relay it to another cloud-connected network (e.g., NB-IoT).

Designing an energy-optimized network with multi-hop capabilities comes with a significant challenge, as each node must constantly listen for network activity to forward data. This can result in high energy consumption. However, achieving operation with a low energy expenditure is essential in remote IoT nodes to ensure prolonged battery life, reduce maintenance needs, and enable efficient deployment in resource-constrained environments.

5.2.1 Related work

Multi-hop LoRa networks have been discussed in the literature. Yet, few authors have focused on energy-efficient tailored network design. Hereby, related works are summarized with respect to the proposed multi-hop network. An overview is presented in Figure 5.2, which is further discussed below.

The comparison between single-hop and multi-hop LoRaWAN networks is presented in [AKA+19]. The authors determine at what range a multi-hop network is needed, and whether the latter is more energy efficient when considering the PDR for various distances, SFs, and transmit powers. They show that to achieve a 90 % PDR, a two-hop network can provide up to 50 % energy reduction as compared to a single-hop network, while increasing 35 % coverage at a particular SF. The authors do not provide a clear and ready-to-use network protocol to implement a reliable LoRa multi-hop network.

5.2.1.1 Synchronous versus asynchronous networks

To support any multi-hop network technology, some form of multiple access solution is required, typically constituting in the MAC layer of a networking protocol. In general, two approaches can be taken with respect to multiple access: time synchronous or asynchronous networks [BDWL10].

Time-synchronized networks transmit time information to all nodes in the wireless sensor network (WSN). These protocols often utilize variants of time-division multiple access (TDMA), where different time slots can be used by different IoT nodes. Common periods such as receive/sleep windows can also be defined. As all nodes are time-synchronous the number of collisions and idle listening time is drastically reduced, yielding an efficient and reliable way of communicating. Yet, setting up and maintaining time synchronization across a vast number of IoT nodes can prove to be challenging, as clock drifts require some sort of recurring time synchronization methods. Hence, as illustrated in [BDWL10], the energy cost of keeping the network synchronized is only permissible in high or medium-load traffic scenarios.

In asynchronous networks, devices do not operate on any schedule or synchronization signal to coordinate their transmissions. In these networks, each device operates independently and starts transmission at a random moment in time. This method can lead to increased collisions and idle listening. Yet, no time synchronization needs to be maintained across the network. This makes this type of network ideal for low-load traffic scenarios [BDWL10].

5.2.1.2 Time synchronized network protocols

Bezunartea *et al.* [BVB+19] adapted a time slotted channel hopping (TSCH) MAC protocol using the LoRa physical layer. They defined the time-slotted protocol to be hopping between three channels during active slots. However, substantial overhead

Figure 5.2: Overview of related work, including the multiple-access technique used, development focus area, and further details of the validation of the protocol in question.

Ref.	Multiple-access Technique	Focus	Experimentally Validated	Firmware		Validation by Simulations	Simulator	
				Open-Source	Platform		Open-Source	Lang.
Asynchronous multi-hop Protocols								
This network	Prolonged preamble sampling	Energy consumption	✓	✓	Arduino	✓	✓	Python
[STD20]	Delay tolerant and remote	Localization	✓	✗	Arduino	✗	-	-
[STB+17]	ALOHA SF differentiation	Energy consumption	✓	✗	LoRaMote	✗	-	-
[ZTDW19]	ALOHA	Coverage	✓	✗	Mbed	✗	-	-
[PMdA+22]	FDMA on LoRa channels	Coverage, throughput	✓	✓	Arduino	✓	✓	Python
[LBP23]	TDOA, SF differentiation	Reliability	✗	-	-	✓	✗	OMNeT++
[ZLS+19]	ALOHA, SF differentiation	Reliability, latency	✓	✗	STM32	✓	✗	?
Time-synchronous multi-hop Protocols								
[PMMB18]	On-demand TDMA	Energy consumption, latency	✓	✗	PIC	✗	-	-
[BVB+19]	TSCH	Coverage	✓	✗	LoRaMote	✗	-	-
[AP19]	Predictive wake-up	Coverage	✗	✗	STM32	✓	-	-
[AFL+19]	Predictive wake-up	Coverage	✓	✗	STM32	✗	-	-
[EGR20]	Predictive wake-up	Energy consumption, coverage	✗	-	-	✓	✗	OMNeT++
[MK20]	TSCH	Latency	✓	✗	Mbed	✗	-	-
[ABBB18]	TDMA	Energy consumption	✓	✗	Zolertia	✗	-	-
[TJY+20; TJYO22]	2-hop slot scheduling TDMA	Coverage, reliability	✓	✗	STM32	✓	✗	?

is required for time synchronization, with renewed synchronization needed every 1000 time slots due to clock drifts. Another multi-hop network was implemented in [AP19], connecting underground environmental sensors in ancient tunnels using a chain-type LoRa multi-hop network with efficient duty-cycling policies that take synchronization errors and delays due to clock offsets and inaccuracies into account. Results show that this mechanism leads to a 50% reduction in energy consumption compared to sub-optimal time synchronization. This technique is also used in [AFL+19] to monitor gas pipelines. Escobar *et al.* [EGR20] implemented a LoRa multi-hop network called JMAC for smart city landscapes, using predictive wake-up times to optimize power consumption and increase the number of devices per cell. Mai and Kim [MK20] proposed a collision-free LoRa multi-hop network that converts the network into a tree topology, assigning timeslots and channels to each link to mitigate the inherent latency disadvantage of multi-hop networks. The IoT nodes communicate using the assigned combination of timeslot and channel, resulting in possible parallel transmissions that decrease the resulting latency. The proposed network was experimentally validated. In [PMMB18], short-range wake-up radios are integrated to lower energy consumption by waking up the IoT nodes by a central coordinator, thus implementing an on-demand TDMA mechanism. An experimental validation shows a PDR of 100% because the on-demand TDMA mechanism eliminates the possibility of packet collisions completely. Tran *et al.* [TJY+20; TJYO22] propose and extend a two-hop real-time LoRa protocol: re-routing is applied to data transmissions via one additional hop to improve coverage and reliability in locations with unfavorable wireless conditions. IoT nodes in the experiments presented in [LBP23] cycle between LoRa SFs to mitigate collisions and improve reliability in the multi-hop network. A new LoRa multi-hop TDMA protocol has been developed by Adame Vázquez *et al.* [ABBB18]: HARE. In the proposed approach, nodes can switch to a multi-hop operation when this is proven to be more energy efficient than single hop operation in classic LoRa point-to-point (P2P) operation. Using HARE, the authors claim an energy reduction for a node of up to 15 %.

5.2.1.3 Asynchronous network protocols

Sciullo *et al.* [STD20] designed an emergency rescue service multi-hop protocol called LOCATE, which uses a personal IoT device to provide LoRa communication to victims where no cellular connectivity is available, with messages re-broadcasted by other peers until reaching a rescue worker who can handle the emergency. Collisions are avoided by assuming a delay-tolerant network in remote environments. Sartori *et al.* [STB+17] propose a novel approach to creating paths in a multi-hop network by using a newly designed MAC protocol called RLMAC. RLMAC is

specifically designed to select the optimal LoRa SF for each available neighbor, enabling nodes to select the path with the lowest time on-air and reducing overall energy consumption. The authors limit their work however to an asynchronous route discovery, without integrating communication possibilities.

In [ZTDW19], the authors improve coverage of LoRaWAN by adding an extra LoRa device in between links, which effectively adds a "hop" between devices. By transferring the ABP credentials to the hopping nodes, communication remains compatible with the LoRaWAN specification. By adding some extra hops in a LoRaWAN network, the PDR and time-on-air (ToA) are improved. Prade *et al.* [PMdA+22] mitigate the challenges of collisions and packet loss in asynchronous multi-hop networks by implementing their "Multi-LoRa" architecture: effectively allowing concurrent transmissions and receptions on multiple LoRa frequency channels. Another strategy, to allow for concurrent transmissions, is the use of the property of the spreading technique used in LoRa, i.e., messages sent with a different SF can be distinguished from each other with negligible interference. This is adopted in [ZLS+19], where each branch of the multi-hop network transmits using a different LoRa SF, allowing for parallel transmissions between branches.

Yet, no open-source asynchronous multi-hop network focuses on ultra low-power operation by exploiting the low data throughput of a remote WSN. The presented multi-hop network aims to fill that gap.

5.2.2 Requirements of the intended applications

Autonomous and remote network– The proposed multi-hop network protocol has to be designed with a "*deploy-and-forget*" strategy in mind, particularly for nodes deployed in remote or hard-to-reach locations. Once a node is properly configured and placed, it should operate autonomously without the need for regular maintenance or monitoring. The multi-hop network needs to support self-configuration and self-healing by (re-)discovering the possible routes.

Low energy consumption – The protocol has to be tailored to nodes operating with a stringent energy budget. Strategies such as reducing the "awake" time of any IoT node, and reducing the number of transmissions by the efficient composition of payloads need to be implemented.

Operation assumptions – To attain these goals, the following assumptions with regard to the applicable use cases are made:

(1) *Sensor measurement data are limited to at most* 12 B. While this payload

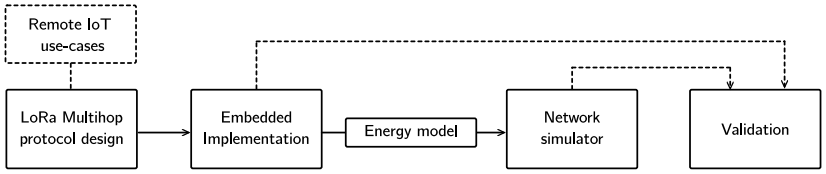


Figure 5.3: Overview of the conducted research: from use case applications to the validation of the presented multi-hop network. The dotted lines represent inputs to the research block in question.

may seem small, this is plenty for the aforementioned use cases. In environmental use cases, for example, this could constitute in a payload of: 4 B temperature value, 2 B humidity value, 4 B specific gas sensing values for estimating the considered air quality. Together, they would fill only 10 B of the payload.

Keeping the data format for sensor measurements short is particularly important to allow for efficient data aggregation. When individual sensor measurement data are kept small, several readings can be aggregated by a 'hopping' node into a single message, before being forwarded.

(2) *Infrequent sensor sampling.* In aforementioned IoT applications, sensor readings only need to take place sporadically. For example, environmental sensors typically sense slow changing quantities, and thus only need to send sensor value readings infrequently (typically at most every 30 min).

(3) *High latency is acceptable.* In logging applications, the data being collected is typically used for long-term monitoring or historical analysis. The data can be used to understand trends or detect patterns over time. In other applications, such as those in smart farming, high latency may be acceptable due to the nature of the measured variables. Smart farming systems are designed to monitor slow-changing environmental parameters, such as temperature, humidity, soil moisture, and light levels. These variables are typically measured over longer periods of time and are not as time-sensitive as other types of data. Action may only need to be taken within a few days based on the collected data. Latency of up to 6 hours is tolerated.

(4) *All data is sent to a central node: the gateway.* The gateway acts as data sink. Also, the network is dense enough so that each node can reach the gateway through one or multiple hops.

5.3 Multi-Hop Low-Power Network Protocol

We propose a multi-hop network that is specifically designed to meet the scope and requirements defined in Section 5.2.2. In this section, we detail the multi-hop network and explain how the presented protocol is able to meet these requirements.

In order to enhance the energy efficiency of sensor nodes, the proposed low-power multi-hop networking protocol incorporates strategies at both the MAC and network layers. At the MAC layer, the detection of prolonged preambles is utilized, as discussed in Section 5.3.1. The devices periodically wake up to check the channel activity. If a prolonged preamble is detected, they process the upcoming payload. Otherwise, the nodes go back to sleep, ensuring that receivers can maximize their sleep time.

To enable energy-efficient communication throughout the multi-hop network and still provide autonomous operation, a routing path is dynamically determined for each node (Section 5.3.3 and Section 5.3.4) at the network layer level. To further improve energy efficiency in this layer, a dynamic aggregation mechanism is put in place, where relay nodes aggregate incoming messages and transmit one large packet rather than several small packets (Section 5.3.5). This way, the energy per effective payload byte in messages is reduced.

5.3.1 Prolonged Preamble Sampling

To optimize the energy efficiency of low-throughput LPWANs, an asynchronous approach outperforms synchronized networks for the use cases under consideration. To mitigate collisions or, in other words, improve the number of successfully received packets, the prolonged preamble sampling technique is used (see Figure 5.4 and Figure 5.5). These preambles are used to detect incoming messages and channel activity. The physical layer protocol LoRa already incorporates preambles to detect and synchronize to an incoming message. However, in the proposed network protocol, the preamble duration in the LoRa message format is prolonged extensively up to 16 s for LoRa SF 7, as opposed to the default value of 6.2 ms for LoRa messages transmitted with SF 7 (250 kHz bandwidth) [13]. This allows us to create an asynchronous multi-hop network without the need for all nodes to be continuously in RX mode. Instead, nodes rely on CAD to receive messages. To ensure activity is detected, receivers wake up at least twice over the time of one preamble transmission to perform CAD (see Figure 5.6). In order to reduce collisions between packets transmitted by neighboring nodes, the time interval between two consecutive CAD operations is randomized. This ensures

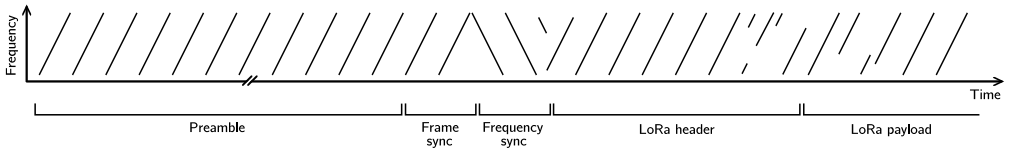


Figure 5.4: LoRa CSS illustration with annotated message fields. The CRC field after the LoRa payload has been omitted in this figure.

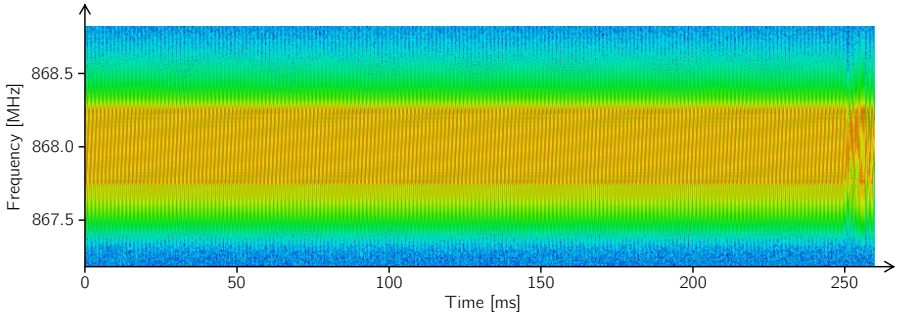


Figure 5.5: Measured waterfall spectrum for a LoRa packet with a prolonged preamble of 250 ms.

that each node begins transmitting at a different moment in time. Otherwise, neighboring nodes that overhear each other's messages could become synchronized. This synchronization could lead to an increase in interference. By introducing randomness in the timing of CAD operations, the network can more effectively avoid collisions.

This method of preamble sampling is increasingly advantageous with respect to energy consumption for WSNs when transmission only seldom occurs, i.e., low-throughput networks. By prolonging the preamble time, less frequent CAD cycles are required, lowering the cost of idle listening. Yet, the longer transmission time of the preambles increases the energy spent in TX mode. This is addressed later in this paragraph. In Section 5.6, we evaluate whether the prolonged ToA of each message results in a higher probability of collisions between messages. The imposed duty cycle limits in the applicable bands are regarded as non-critical, because in this protocol, we focus on highly infrequent data transmissions.

The trade-off between energy consumption and preamble duration is depicted in Figure 5.7 and is highly dependent on the transmit interval. This trade-off is explored by calculating the lifespan of a relaying node for varying preamble

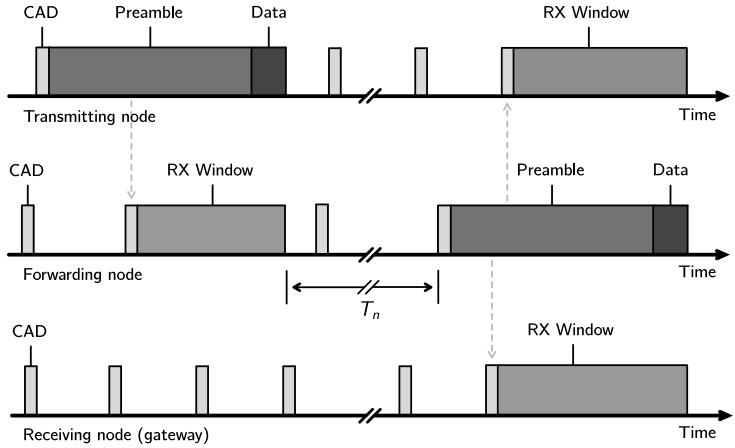


Figure 5.6: Illustration of how the prolonged preamble sampling technique can be employed in a multi-hop network.

duration.[2] In this calculation, we adopt the assumption that each node forwards every received message exactly once, acting as an intermediate forwarding node in the multi-hop network. Furthermore, if the node has sensor data available, that data is assumed to be appended to that incoming message before it is forwarded. In essence, a single outgoing message is sent for each incoming message. Additionally, we assume a perfect transmission scenario with no packet loss. The lifespan of the node incorporates the total energy consumption: sleep, transmit, CAD and receive energy consumption [TCLW19]. A clear optimum can be observed for each message interval, e.g., a preamble duration of 3.84 s when sending and receiving messages every two hours. In the presented multi-hop protocol, the IoT nodes perform a CAD at two random times during this preamble duration of 3.84 s. This calculation is further used as an estimation of the most optimal preamble duration for the presented multi-hop network.

The CAD mechanism, implemented in LoRa modems, is a very power-efficient manner to check the channel for activity. In CAD mode, the modem does a fast scan of the band to check for a LoRa preamble. First, the internal PLL locks on the channel frequency. Secondly, the receiver records the channel for the time of one symbol length. In this phase, the receiver consumes as much power as in the RX phase, however only for the duration of one LoRa symbol:

$$T_{\text{sym}} = \frac{2^{\text{SF}}}{\text{BW}} \ , \tag{5.1}$$

[2] Calculator tool available online and open-source: dramco.be/tools/lora-multihop

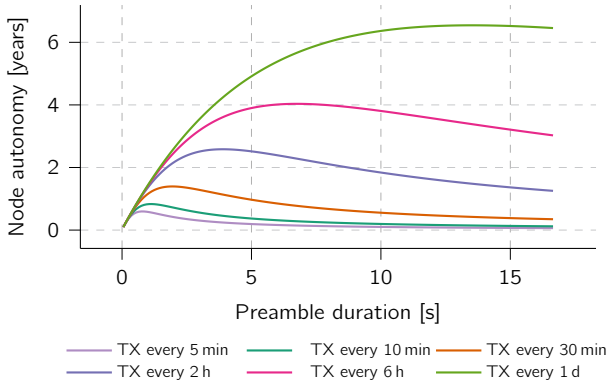


Figure 5.7: Exploration of the lifespan of a multi-hop sensor node versus the set preamble duration. Calculations based on equal amount of RX and TX, SF7, 500 kHz bandwidth, and 30 B payload. The chosen battery size is equal to the capacity of two common AA batteries: 2500 mAh. Note that this assessment does not incorporate the self-discharge rate of the batteries.

where SF represents the chosen LoRa spreading factor and BW the applicable LoRa bandwidth. In a network that is configured with SF = 7 and BW = 500 kHz, the symbol time T_{sym} would equate to 0.26 ms.

Finally, the digital processor inside the modem determines the cross-correlation between the channel recording and the ideal preamble waveform. This process takes less time than one symbol, and the power consumption is reduced in this phase. If channel activity is detected, an interrupt is given to the application processor, after which the modem goes back to sleep.

As the CAD mechanism occurs rather frequently, it is important to keep its energy consumption, and thus duration, to a strict minimum.

When comparing the lifespan of an IoT node that uses prolonged preamble sampling versus continuously listening for incoming packets, a large improvement on node autonomy is achieved. When comparing the aforementioned values (e.g., sending a message every 2 hours on two AA batteries), and considering the energy profile of the hardware (as obtained later in Figure 5.13b), a lifespan of 2.5 years can be achieved. By opening a continuous RX window, the lifespan shrinks to only 2 days.

5.3.2 Frame structure

In order to enable routing and aggregation in the proposed multi-hop network, a new frame structure[3] must be introduced. This new frame structure is used for all messages in the network (refer to Figure 5.8). The message header, which is the first part of the frame, is added before the application payload. The message header has a fixed size of 7 bytes[4] and includes the following fields:

- `Msg UID`: The first 16 bits are a randomly generated message identifier. This is used by the nodes to detect duplicate messages.
- `Msg type`: This byte specifies the type of the message. Currently, three types are supported: (1) `ROUTE_DISCOVERY` (2) `ROUTED_DATA`, en (3) `DATA_BROADCAST`.
- `Hops`: The number of hops a message has taken to arrive at a certain node. This field is automatically updated by the nodes when forwarding a message and plays an important role in the route establishment phase.
- `Cumulative LQI`: A 16-bit number giving an indication on the quality (hence, link quality indicator (LQI)) and reliability of a certain route in the network (see Section 5.3.3). In the current implementation, a lower number indicates better quality. When a node forwards a packet it increases the cumulative LQI field with the LQI of the previous hop.
- `Addr`: an address field used both in routing and route establishment. For routed data packets, this field contains the destination address. For route discovery packets, this field contains the source address of the transmitting (forwarding) node.

This header consisting of five fields is followed by the payload. The payload is further subdivided into payload blocks, which can be nested into each other as a result of the aggregation algorithm. Each payload block consists of the data sent by each sensor node, accompanied by a small 3-field payload header:

- `Src UID`: an 8 bit[5] unique identifier of the sending node.
- `L1`: length of the sensor data generated by this node.
- `L2`: length of the data that is generated by other nodes, but is forwarded by this node.

[3] The frame structure refers to the contents of the LoRa payload, as depicted in Figure 5.4.
[4] The header is 8 bytes long when 16-bit addressing is being used.
[5] The `Src UID` is 16 bit long, when 16-bit addressing is used in the network.

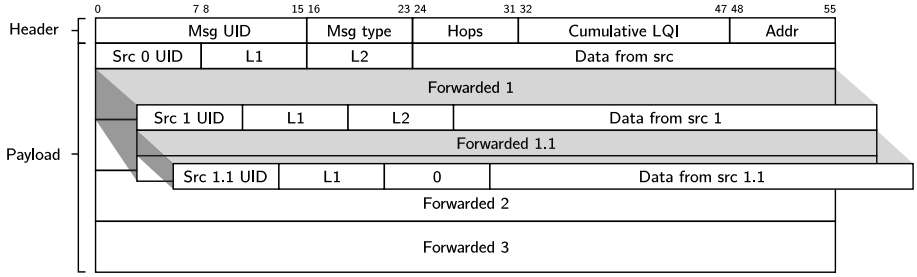


Figure 5.8: General message structure. The header with five fields (7 B) is followed by the payload. Depending on the type of packet and a node's position in the network, the payload can have a nested structure.

The length of the payload varies according to the payload bytes required from the general use case (`L1`) and the node's position in the network (`L2`). Thus, the payload data is subdivided in two parts:

- `Data from src`: data that is generated by the node considered itself, the length of this data is indicated in field `L1`.
- `Forwarded data`: incoming data that needs to be forwarded to other nodes, constituting in a single "hop". The total length of the forwarded data is recorded in field `L2`. The forwarded data field consists of a nested structure of several payload blocks, after several hops. Each payload block is kept distinguishable by a payload header, which incorporates the appropriate lengths.

The above proposed general message structure is optimized to ensure proper operation of the multi-hop network, as is further validated.

5.3.3 Route Establishment

In multi-hop networks, message delivery to a central gateway can be organized in two ways: by flooding a message throughout the network, or establishing routes to the gateway for all nodes [AWW05]. As we consider mostly static networks and optimize for energy efficiency, it is obvious to implement a route establishment protocol [BSMP11; DG18]. This method should ensure message delivery in the least amount of hops, and opt for a good quality wireless link between hops.

With predetermined intervals, the gateway transmits a `ROUTE_DISCOVERY` message. Every sensor node that receives this message, re-transmits the package. By doing

so, the ROUTE_DISCOVERY propagates through the whole network. By checking the Msg UID, nodes ensure that they transmit a received ROUTE_DISCOVERY packet only once. When a node re-transmits the ROUTE_DISCOVERY message, it adjusts only three fields in the message header: replacing the Src UID with its own UID, incrementing the hop count (Hops field) by one and adjusting the Cumulative LQI field accordingly. The latter is calculated as follows for each route $\mathcal{R}_n$:

$$\mathrm{LQI}_{\mathcal{R}_n} = \sum_{l \in \mathcal{R}_n} \mathrm{SNR}_{\max} - \mathrm{SNR}_l, \tag{5.2}$$

where the set $\mathcal{R}_n$ contains the links or hops l between the gateway and node n, $\mathrm{SNR}_{\max}$ depicts the maximum SNR by the LoRa modem and SNR_l describes the SNR over the link l. The SNR values are expressed in dB. The maximum SNR ($\mathrm{SNR}_{\max}$) with the current hardware considered in this paper is 30 dB.

Every sensor can receive the same ROUTE_DISCOVERY message several times, i.e., from all its neighbors, and thus each representing a possible route to the gateway. For each incoming ROUTE_DISCOVERY message, the three most important fields get saved in a circular buffer data structure (i.e., the routing table) of the last eight observed routes: Src UID, Hops, and Cumulative LQI. From this list, the most suited route is chosen for all future ROUTED_DATA messages, until a new ROUTE_DISCOVERY message is received. The route with the lowest Cumulative LQI is chosen as the most suited route. When the Cumulative LQI is equal for two or more routes, the route which entails the least amount of hops is chosen. This proposed route establishment mechanism ensures that routed messages are sent to the gateway along the links with the highest accumulated SNR, thereby mitigating packet loss due to low signal strength. Furthermore, Eq. (5.2) ensures that routes are taken with as few hops as possible. Lowering the number of hops and favoring high-SNR links, fewer message forwards and retransmissions are required, improving the energy efficiency of the network.

To clarify the above process, consider the network, shown in Figure 5.9, which includes one gateway with UID *00* and four sensor nodes with randomized UIDs. The gateway initiates the network by sending a ROUTE_DISCOVERY message. Upon receiving this message, nodes *01* and *02* update their routing table, and then forward it after a random delay between $\Delta T_{\min}$ and $\Delta T_{\max}$. In the example, node *02* forwards the message first, after updating the Addr, Hops, and Cumulative LQI fields accordingly. Of the nodes that receive this forwarded message, only nodes *03* and *04* further forward the message, as node *01* already forwarded the original ROUTE_DISCOVERY message. Node *01* only uses the incoming message to update its routing table. Any subsequent messages received by nodes *01*, *02*, *03*, and *04* are not forwarded as all receiving nodes have already processed

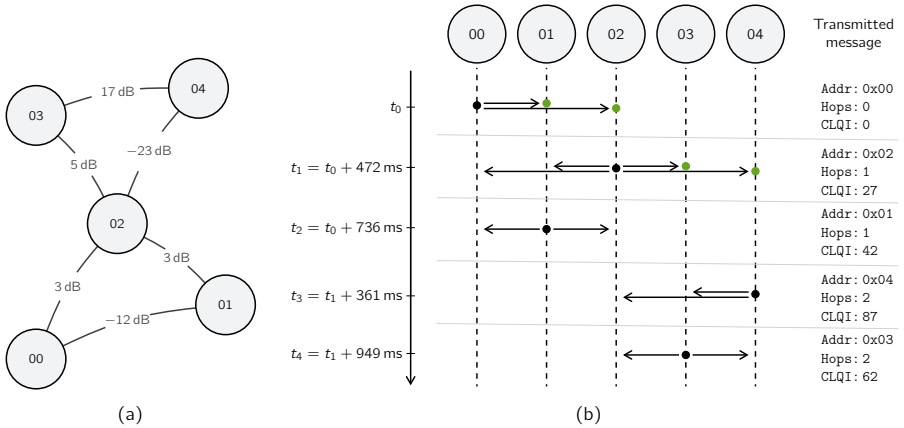


Figure 5.9: Example of the route discovery mechanism. **(a)** Example network topology, the SNR value of each link is depicted between IoT nodes. **(b)** Timeline of the route discovery messages propagation through the network. Each forwarded message (i.e., hop) occurs at a random time ($\mathcal{U}[\Delta T_{\text{min}}, \Delta T_{\text{max}}]$) after message reception. Only relevant message fields of the transmitted messages are shown.

the `ROUTE_DISCOVERY` message. The incoming messages are used to update their routing tables accordingly. The resulting routing tables are depicted in Figure 5.10a.

It is clear that routes in the network remain static between `ROUTE_DISCOVERY` messages sent by the gateway. The timing of sending these messages from the gateway can be chosen according to the dynamism of the network surroundings: more frequent for changing urban landscapes, and less frequent for more static rural environments.

5.3.4 Message Routing

Routed messages navigate across the multi-hop network to the gateway, according to the route established by the route discovery protocol (described in Section 5.3.3). When latency tolerant data is available on a sensor node, it can be sent using `ROUTED_DATA` messages. Each of these messages contain the unique identifier of the node that needs to forward this message in the `Addr` field of the message (from the routing table). When a node gets a `ROUTED_DATA` message, it checks the `Addr` field against its own `UID`. When these match, the `Addr` field is updated according to the routing table of the forwarding node and the `Hops` field is incremented by one before re-transmitting the incoming message. Other `ROUTED_DATA` messages

01

Addr	Hops	CLQI
00	0	42
02	1	54

02

Addr	Hops	CLQI
00	0	27
01	1	69
03	2	77
04	2	133

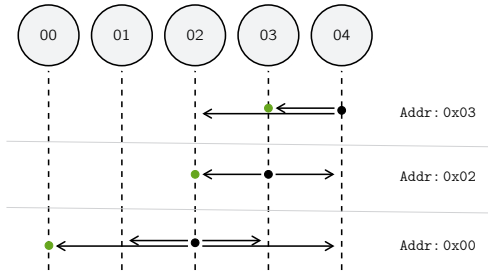
03

Addr	Hops	CLQI
02	1	52
04	2	93

04

Addr	Hops	CLQI
02	1	80
03	2	65

(a)



(b)

Figure 5.10: Example of how data is routed to the gateway, using the network topology from Figure 5.9a. **(a)** The resulting routing tables of each sensor node in the network, the best route is selected by the lowest `Cumulative LQI` field, as highlighted. **(b)** Routing of a message sent by sensor node 04. Only the node with the appropriate UID (i.e., `Addr`) will forward the message until the gateway 00 is reached.

are ignored, thus each `ROUTED_DATA` message is only forwarded once per hop level. To improve the overall energy efficiency of the multi-hop network, nodes can delay the re-transmission of incoming `ROUTED_DATA`. Messages that arrive during this extra delay can be merged, hence improving the spent energy per byte.

To clarify, consider the network and example mentioned in Section 5.3.3 and Figure 5.9. Node 04 sends data via the multi-hop network to the gateway. The routing table of node *04* states that the most suited route starts with node *03*: the sent `ROUTED_DATA` message is addressed to node *03* (i.e., `Addr` field). Other nodes ignore the message. After waiting for possible aggregation opportunities (as described in Section 5.3.5), node *03* adjusts the `Addr` field accordingly and forward the message to the next "hop". This mechanism continues until the gateway (*00*) is reached.

5.3.5 Aggregation to Improve Energy per Byte

One way to reduce the energy cost of the preamble and header transmission is through the use of aggregation techniques [CLV+21]. This can greatly improve the energy efficiency of the system, especially for applications with a high overhead-to-payload ratio. This is the case for the presented multi-hop network, because of the usage of the prolonged preamble at the beginning of each message.

The effect of payload aggregation on the energy consumption by transmission per transmitted sensor reading is displayed in Figure 5.11. It is clear that energy

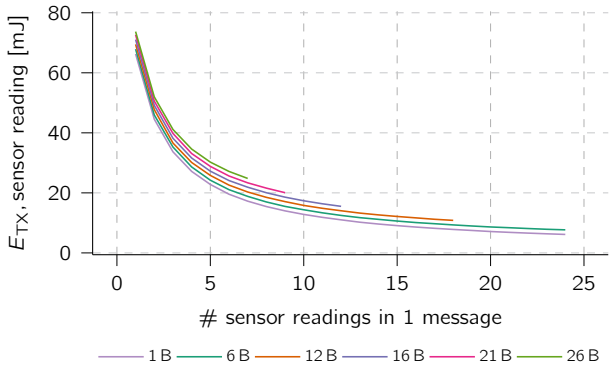


Figure 5.11: Effect of aggregating multiple sensor readings in one message on the energy consumption. Calculated for multiple sensor reading sizes, limited by the maximum LoRa message size (255 B), and with a 1 s preamble length.

consumption per sensor reading, for the assumed data length per reading, can be reduced by a factor of seven by fully making use of the available LoRa payload size of 255 B.

Multiple aggregation opportunities with regard to aggregation emerge when observing the workings of the multi-hop network. A relaying node receives sensor readings from multiple sensor nodes and thus is able to aggregate incoming sensor readings as well. In the proposed network protocol, this is implemented with a dynamic aggregation timer: after an intent to transmit, a node waits for any incoming messages that can be appended to the message currently in the queue to be transmitted. To fully leverage the benefits of the aggregation timer, the timer value should adapt dynamically to maximize data aggregation within a single packet. This entails extending the timer duration when additional extra data can be appended to a pending message and only shortening it when the message reaches full capacity. However, unwarranted latency for nodes that do not act as data forwarders, such as leaf nodes, should be mitigated. Therefore, in cases where no messages are received throughout the timer's duration, the timer should also be reduced.

The aggregation timer begins after the occurrence of a transmission intent (denoted by t_n, with n being the nth transmit cycle), either due to sensing or receiving a `ROUTED_DATA` message, and expires after the specified time $T_{a,n}$ known as the aggregation timer delay. During this time, incoming messages are accumulated and combined into a single message buffer by appending each message's payload. The aggregated message is then transmitted when the timer expires or the

buffer becomes full. Despite its energy-saving benefits, aggregation can result in significant latency in multi-hop wireless sensor networks. As a result, the duration of the aggregation timer adjusts based on network traffic, increasing by $T_{a,\text{upstep}}$ with each incoming message, up to the limit of $T_{a,\text{max}}$. When no messages are received or the buffer is full, the timer decreases by $T_{a,\text{downstep}}$ for the next aggregation cycle, down to the minimum value of $T_{a,\text{min}}$. Thus, the timer duration is computed as,

$$T_{a,n} = \begin{cases} \max\left(\left(T_{a,n-1} - T_{\text{a,downstep}}\right), T_{\text{a,min}}\right), & \text{if } M_{n-1} = 0 \text{ or full buffer} \\ \min\left(\left(T_{a,n-1} + M_{n-1}\, T_{\text{a,upstep}}\right), T_{\text{a,max}}\right), & \text{otherwise} \end{cases} \tag{5.3}$$

with M_{n-1} the number of received messages during the previous aggregation period $T_{a,n-1}$.

The timer starts at the intended transmission time t_n, and ends at $t_{\text{exp},n}$ (as defined by Equation (5.4)). A randomized delay, drawn from a uniform distribution (denoted $\mathcal{U}$) between $-\Delta T/2$ and $\Delta T/2$, is also added for every node before transmission.

$$t_{\text{exp},n} = t_{\text{n}} + T_{a,n} + \mathcal{U}\left[-\Delta T/2, \Delta T/2\right] \tag{5.4}$$

This additional delay mitigates the chance of simultaneous transmissions by nodes that have the same aggregation time, hence avoiding collisions and packet loss.

An example timeline on the use of the dynamic payload aggregation model is depicted in Figure 5.12. The node starts a new message by sensing a sensor and starts the first aggregation timer $T_{a,n}$ (offset by a randomized delay). During $T_{a,n}$, two `ROUTED_DATA` messages are received that need to be forwarded by the node. When the aggregation timer expires, all payloads get merged in one message that is transmitted. As this message contains two extra payloads, the aggregation timer value that is used in the future $T_{a,n+1}$, gets extended two times by $T_{a,\text{upstep}}$. Next, the node starts a new aggregation timer once another `ROUTED_DATA` messages is addressed to it. The timer is, again, offset by a random value. As, in this case, no extra messages are received to append, the next aggregation timer $T_{a,n+2}$ is reduced by one $T_{a,\text{downstep}}$.

5.4 Multi-Hop Open-source IoT Platform

To accurately evaluate the performance of the network in practical scenarios, we have implemented the presented multi-hop protocol in a network stack, developed

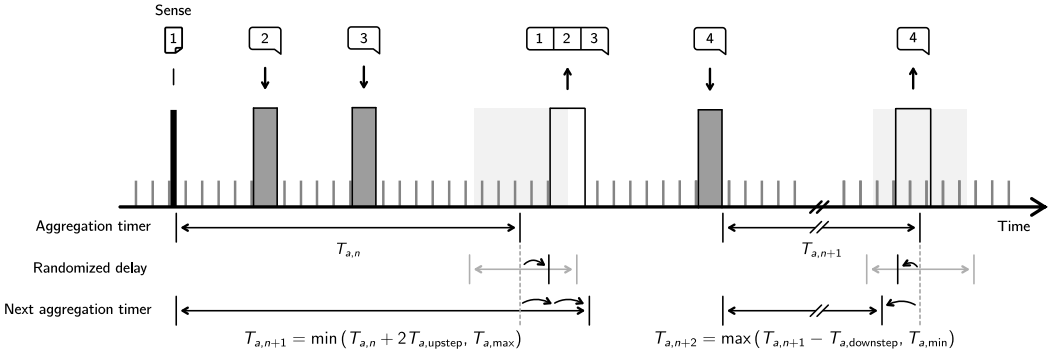


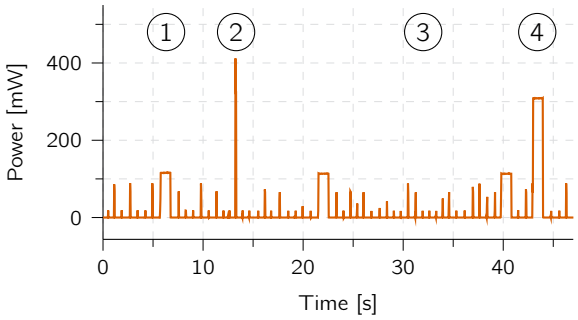
Figure 5.12: Payload aggregation model as implemented in the presented multi-hop network: minimizing energy consumption by utilizing payload aggregation and minimizing the resulting latency by varying the aggregation timer based on the network throughput.

for Arduino-powered embedded systems. The employed hardware [TCLW19] and firmware used are both available and open-source.[6]

The network stack implementation is designed to run on the "Dramco Uno" platform [LOC22], which is a modification of the "Arduino Uno", with an added on-board LoRa modem and extra energy-saving measures. This custom board features the Microchip Atmega328p microcontroller and Semtech SX1276 LoRa transceiver. The selection of these components was based on their low cost, availability, and ease of development. The hardware platform is described in detail in [TCLW19]. The use of these components allows the firmware to be run on a variety of other platforms without modification. Arduino-based platforms that use the SX1276 can also be targeted with little or no modification to the firmware.

The stack implementation makes use of various available energy-saving techniques, including deep sleep techniques. A fine-grained power model of this hardware was recorded and is available in Figure 5.13b. This model includes low-power sleep consumption (23 µW) and periodic CAD cycles (330 µJ), leading to an average idle power consumption of 2.1 mW (when using SF 7, a LoRa bandwidth of 250 kHz, and a preamble duration of 1 s). These measurements are used across this publication to calculate various energy-saving impacts and the lifespan of the IoT nodes.

[6] The firmware is available as open-source at github.com/DRAMCO/LoRaMultiHop.



(a)

	Mode	Power [mW]
	Deep sleep	0.023
(3)	CAD cycle	39.1
(1)	RX mode	116.1
(4)	TX mode	309.2
(2)	Sensing	320.4

(b)

Figure 5.13: Measured power profile of the implemented LoRa multi-hop network, operating at SF 7. These stages can be observed, alternated with deep sleep: (1) RX mode, (2) sensing mode, (3) repetitive CAD cycles, and (4) TX mode.

5.5 Multi-Hop Network Simulator

To evaluate and optimize network parameters under various conditions, we developed a discrete-event cross-layer simulator in Python based on LoRaEnergySim [COv19]. The simulator is available as open-source.[7] It models the behavior of IoT nodes (e.g., sensing, energy consumption profile) and implements the proposed protocol. The overall topography of the simulator is depicted in Figure 5.14. In the framework, a *sensor* node initiates a *message* (containing the gathered sensor data). The message is sent via the *link* interface to the surrounding nodes. The message proceeds through the network according to the *route* protocol (see Section 5.3.4), before arriving at the *gateway* node.

5.5.1 Nodes

A node can be one of two types: *sensor* or *gateway*. A sensor node actively takes part in sensing and forwarding sensor data in the network. The gateway operates as a data sink: all collected sensor readings are stored here.

The operations of each node are characterized by a set of properties and classes (as illustrated in Figure 5.14): (1) network-related parameters, (2) energy profile, (3) node location, (4) protocol timers, and (5) discovered route.

[7] Available at: github.com/DRAMCO/LoRa-multihop-simulator

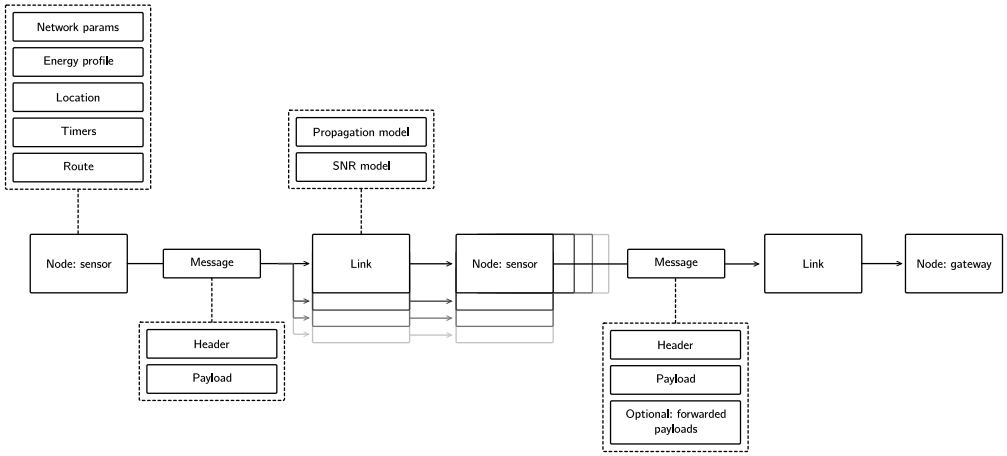


Figure 5.14: Overview of the network simulation framework structure, used to evaluate the proposed protocol

1. **Network related parameters.** The specifications of the network behavior are collected in a single class. These include both LoRa parameters (e.g., SF, sensitivity) and multi-hop protocol parameters (e.g., preamble size, route buffer size). The resulting changes in network behavior can be simulated by iterating through a range of network property values

2. **Energy profile.** The energy profile is depicted in Figure 5.13b. Six states are used in the firmware implementation of the protocol: sleep, CAD, RX, TX (CAD), TX (data), and sensing. The duration and consumed power in each of these states is accurately measured and included in Figure 5.13b.

3. **Node location.** Each node is put into the network on a specific 3D location. Link properties such as the euclidean distance between two nodes, especially, are calculated based on these values. Other link parameters (e.g., RSSI and SNR) are derived from the distance in the Link class.

4. **Protocol timers.** The proposed network protocol incorporates multiple timers. All timers are pre-configured based on the list of network-related parameters. Both the sensor and gateway nodes use a *sleep timer*, to define the CAD intervals. One additional timer is running on the gateway node, i.e., the *route discovery timer*. This timer indicates when a route discovery message needs to be distributed in the network. For sensor nodes, a total of three extra timers are running: *sense timer*, *collision timer*, and *aggregation timer*. The *sense timer* indicates when the node's onboard sensors values need to be collected. The *collision timer* mitigates collisions

by including a back-off window after a CAD cycle determines channel activity. The *aggregation timer* controls the opening and closing of the aggregation window to gather data from multiple nodes in one message.

5. **Discovered route.** Based on the route discovery protocol, each node determines which should be the destination node, i.e., which node should be the recipient of all the messages sent by the node. This destination node and the associated LQI are stored in the *Discovered route* class. Whenever a new route discovery message is received, this message is passed to the *Discovered route* class to update the destination node.

5.5.2 Messages

Messages in the simulation framework adhere to the fields of the presented network protocol, as depicted in Figure 5.8: incorporating both header fields and payload lists. When nodes forward a message, forwarded payloads get copied into a payload list in the simulator (similar to Figure 5.8). To be able to track each message from node to node, each payload in the simulation framework is extended with a list of the identifiers of each forwarding node. This allows the simulator to identify the exact path and thus the exact number of hops a payload needs to travel before arriving at the gateway. Importantly, this extension has no impact on the network's operations and serves solely as metadata.

5.5.3 Links

Separate link classes exist for each combination of two nodes, which offers the interface of two nodes to send and receive messages. This class consists of two main components: (1) a propagation model to introduce the path loss (PL), and (2) an SNR model to include noise.

1. **Propagation model.** The simulation framework offers –by default– three channel models, based on measurements obtained in multiple environments: open/coastal, forested, and urban environments from [CV20].[8] They are based on a log-distance PL model with a per-link shadowing factor induced by e.g., blockage. The log-normal path loss PL at a distance d in 2D space

[8]Note, that other PL models can be easily implemented in the framework, and thus the framework is not limited to the here proposed models.

can be expressed as:

$$\mathrm{PL}(d) = \mathrm{PL}(d_0) + 10\rho \log\left(\frac{d}{d_0}\right) + \mathcal{X}_\sigma \ , \tag{5.5}$$

where, d_0 is the reference distance and thus the minimal distance between transmitter and receiver, ρ the path loss exponent and $\mathcal{X}_\sigma$ models the shadowing component. The shadowing component is modeled as a zero-mean Gaussian distributed random variable with standard deviation σ, $\mathcal{X}_\sigma \sim \mathcal{N}(0, \sigma)$.

The parameters for the different environments are summarized in Table 5.1. The considered PL models deviate from conventional PL models proposed by 3GPP and others [RLHH21; PMR+15a], as the network topology differs from the assumed topologies, i.e., the nodes are commonly placed on the same height rather than having an elevated height for the gateway as assumed in other work. It is important to recognize that the path loss component can vary significantly depending on its situation within a real-life scenario. Moreover, it exhibits dynamic characteristics over time due to its sensitivity to a wide range of environmental parameters such as atmospheric conditions (rain or humidity) and obstructions.

Table 5.1: PL parameters for each considered environment (based on [CV20])

Environment	d_0 [m]	$\mathrm{PL}(d_0)$ [dB]	ρ	σ [dB]
Open/coastal	1	43.96	3.62	27.51
Forested	1	95.52	2.03	6.87
Urban	1	74.85	2.75	11.25

2. **SNR model.** In the standard operation, the simulator framework takes thermal noise [17] into account to compute the SNR of the signal, i.e.,

$$\mathrm{SNR}_{\mathrm{dB}} = \mathrm{TP} - \mathrm{PL} - N \ , \tag{5.6}$$

with TP the transmit power (in dBm) and N the AWGN power (in dB). The noise power is computed as $N = k_\mathrm{B} T \Delta f$ with k_B the Boltzmann constant, T the temperature and Δf the signal bandwidth.

In future versions, more complex link models can be included. For example, the LoRa simulator presented by Al Homssi *et al.* in [ADM+21] includes an accurate interference algorithm and channel emulator which employs ray-tracing to accurately model the effect of multi-path fading.

5.5.4 Output

For any set of network parameters, the following resulting metrics can be simulated.

- **Packet delivery ratio**: The framework provides both the overall and per-node PDR w.r.t. a node's position in the network. This takes both the path loss model (described in Section 5.5.3) and potential collisions with transmissions from neighboring nodes into account.
- **Energy models**: The simulator uses the in-depth energy profile of the used hardware. As a result, energy-related metrics can be simulated: energy consumed by one node (w.r.t. the position in the network), energy load across the network and the energy consumed per packet (and thus per byte or per payload).
- **Latency**: The time between the intent of sending a sensor value and the time of receiving the data on the gateway (i.e., latency) is recorded on a per-payload basis.
- **Aggregation ratio**: The aggregation ratio $\alpha_{\text{aggregation}}$ is defined as the ratio of the messages sent by one node that included payloads from other messages ($m_{\text{aggregated}}$) to the total number of sent messages (m_{total}):

$$\alpha_{\text{aggregation}} = \frac{m_{\text{aggregated}}}{m_{\text{total}}} \quad . \tag{5.7}$$

The framework enables simulation of the impact of various network parameters (including node positions, network density, and payload size) on the metrics mentioned. This is achieved through Monte Carlo simulations, where a set of randomly selected network values within a given distribution is repeatedly simulated to observe the effects on the metrics.

5.6 LoRa Multi-Hop Evaluation

The proposed network protocol was analyzed by conducting two experiments. First, we evaluate the effects on the energy efficiency of the proposed aggregation methods, in a real-life demonstration network. In our second experiment, by using the simulation framework, we are able to establish an ideal set of network settings (w.r.t. energy efficiency) for any use case. These parameters get validated experimentally in a local campus-wide test.

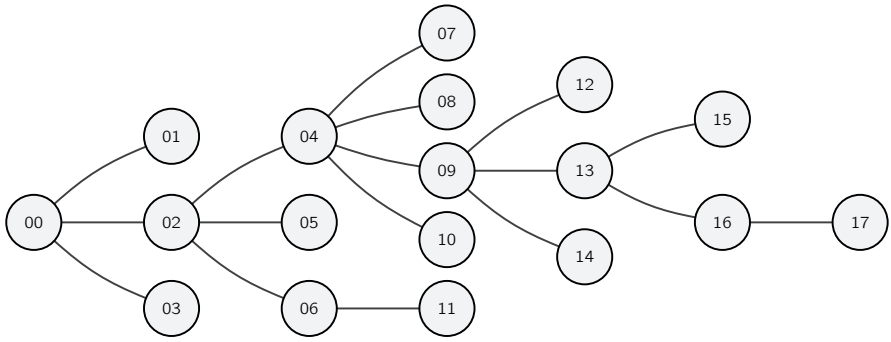


Figure 5.15: Fixed network topology (as followed by the routing mechanism) to study the impact of aggregation across the multi-hop network.

5.6.1 Experiment 1: Impact of Performing Aggregation in a Multi-hop network

The implemented aggregation feature in the proposed multi-hop network is expected to reduce energy consumption and increase data latency. We experimentally study its impact on these parameters. To ensure consistency, we established a fixed network topology and configuration: an 18-node IoT network with a randomly generated topology, as depicted in Figure 5.15, overriding the automatic route discovery mechanism. Other relevant network parameters are listed in Table 5.2.

Table 5.2: Network parameters used to study the impact of aggregation across the multi-hop network

Parameter	Symbol	Value
Measure interval	T_{measure}	10 min
Minimal aggregation timer	$T_{a,\text{min}}$	0 s
Initial aggregation timer	$T_{a,0}$	2.5 min
Maximal aggregation timer	$T_{a,\text{max}}$	5 min
Aggregation upstep	$T_{a,\text{upstep}}$	1 min
Aggregation downstep	$T_{a,\text{downstep}}$	30 s
TX buffer size		150 B
Payload size		6 B
LoRa spreading factor	SF	7
LoRa bandwidth	BW	500 kHz
Preamble width	T_{preamble}	1 s

5.6.1.1 Reducing Energy Consumption by Aggregation

We evaluate the aggregation ratio and resulting energy savings experimentally. The results for relaying nodes are summarized in Table 5.3. We use two metrics to compare the energy consumption of the network. $E_{B,\text{no aggregation}}$ represents the average energy consumption per payload byte when aggregation is disabled in the network, while $E_{B,\text{aggregation}}$ indicates the energy consumption per payload byte when aggregation is active in the network.

In our study, nodes *06* and *16* exhibited relatively low aggregation ratios, with approximately half of their transmitted messages containing forwarded payloads. This is expected, as these nodes have only one child node sending data to them, which limits the potential for aggregation. Nodes *02* and *04* demonstrate a high aggregation ratio, which leads to substantial energy savings when aggregation is enabled for the network. Our results confirm that energy savings are most pronounced for nodes with a high number of children. Specifically, in our study, the sensor node that forwards data of four children is able to reduce its energy consumption by up to 61 % per transmitted message by leveraging data aggregation.

Table 5.3: Impact of the proposed aggregation mechanism on the energy consumption of relaying nodes in the network, depicted in Figure 5.15 (which is using the configuration noted in Table 5.2).

Node UID	# children	$\alpha_{\text{aggregation}}$	$E_{\text{TX},B,\text{no aggregation}}$[mJ]	$E_{\text{TX},B,\text{aggregation}}$ [mJ]	Savings
02	3	96%	52.40	35.53	32%
04	4	92%	52.40	20.48	61%
06	1	48%	52.40	48.20	8%
09	3	89%	52.40	33.27	37%
13	2	77%	52.40	43.02	18%
16	1	51%	52.40	43.73	16%

5.6.1.2 Reducing Latency by Dynamic Aggregation

Increased latency is an inherent drawback of using aggregation in a multi-hop network. To combat adding unnecessary delays to nodes that do not forward data due to their position in the network, a dynamic timer mechanism is put in place (see Section 5.3.5). To validate this mechanism, several nodes in the fixed network depicted in Figure 5.15 are logged to keep track of timer value variations. The evolution of the aggregation time T_a is depicted in Figure 5.16. Both axes

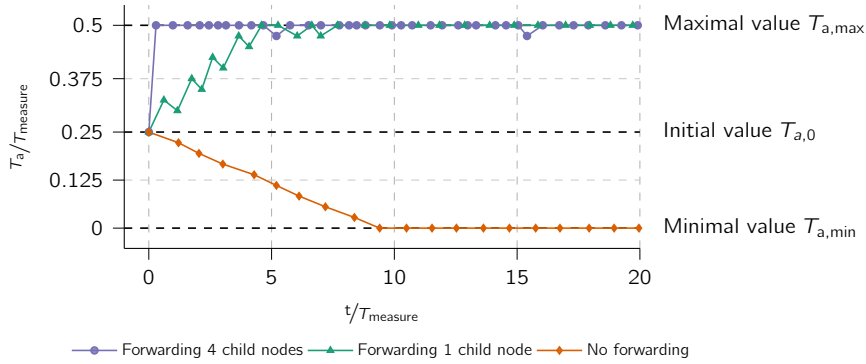


Figure 5.16: Experimentally validated convergence of the aggregation timer, normalized to the measurement delay $T_{measure}$.

have been normalized to the measurement delay $T_{measure}$, as the aggregation timer T_a varies depending on the availability of data (and thus the measurement interval $T_{measure}$).

It is clear that all nodes eventually opt for either the maximal aggregation delay $T_{a,max}$ (maximizing aggregation ratio $\alpha_{aggregation}$) or the minimal $T_{a,min}$. Nodes that receive a lot of incoming data during the aggregation window, quickly saturate on $T_{a,max}$. This stems from the fact that the protocol has been designed with energy efficiency in mind. Nodes that forward data, need to increase their chance of accumulating data. This is what happens, for example, with node *04* in Figure 5.15. This node forwards data from four other child nodes: $T_{a,max}$ is reached after only one measurement interval $T_{measure}$. Other nodes which, for example, only forward data from one child node (such as node *06* or *16* in Figure 5.15) need more time to saturate to the maximal aggregation timer value $T_{a,max}$. Nodes that do not forward data, i.e., leaf nodes, never aggregate data. Consequently, their aggregation timer evolves to the minimal aggregation timer value $T_{a,min}$ and minimizes latency from the leaf node to the gateway.

To quickly maximize the aggregation potential, the T_a is increased with a step size of $T_{a,upstep}$. To keep it maximized for nodes with a lower aggregation ratio, the decrease of T_a, denoted $T_{a,downstep}$ is taken smaller as $T_{a,upstep}$ (see Table 5.2 for the values used). This can also be seen in Figure 5.16. However, when a node stops forwarding data due to a change in the routing, e.g., because of a change in the environment or the network itself, its T_a would evolve to $T_{a,min}$ over time. Our results (see Section 5.6.2.1 and Figure 5.20) show that this mechanism indeed optimizes the energy consumption of the aggregating nodes.

5.6.2 Experiment 2: University campus deployment

The proposed multi-hop network is evaluated based on a series of experiments in real-world settings. We deployed the experimental platform across the university campus, taking measurements and collecting data to assess the network's performance under various scenarios. However, to identify the most optimal network settings prior to deployment, we first conducted simulations using the network simulator. By running simulations with different configurations and traffic loads, we were able to evaluate the network's behavior and identify the best settings that would maximize its performance while minimizing energy consumption and other resources. This is discussed in Section 5.6.2.1.

The deployed network is depicted in Figure 5.17. In total, 30 nodes and one gateway are deployed across campus, both indoors and outdoors, comprising a total area of 200 m by 170 m with some indoor sensor nodes placed at different floors. The LoRa parameters are set to the settings providing the lowest range, i.e., output power of 0 dBm and SF 7. In this way, we intentionally emulate a more harsh propagation environment, similar to the conditions present in a more large-spread deployment. The range is further reduced by placing most of the nodes indoors. This is to stress-test the proposed protocol and illustrate its applicability to the mentioned target applications.

5.6.2.1 Determining Optimal Configuration via the Simulator

Using the simulation framework described in Section 5.5, we simulate the presented LoRa multi-hop network (Figure 5.17) to explore the impact of various parameters. All links are simulated using the urban path loss model (as noted in Table 5.1). Criteria on the performance of any parameter combination are the resulting PDR, packet loss ratio (PLR) due to collisions, energy consumption, and latency. In particular, we simulate the effects on the performance of the network of the following parameters: (1) network throughput, (2) aggregation time, (3) randomization parameters, and (4) payload size.

1. **Network throughput.** In this implementation of the multi-hop WSN, the size of generated sensor data is fixed for all nodes in the network. As a result, network throughput is highly dependent on the rate at which data is generated by the IoT nodes. This rate is determined by the measurement timer T_{measure} used to capture sensor data. Lower values of T_{measure} lead to higher throughput, possibly saturating the network due to a large number of collisions caused by a longer ToA per packet, as dictated by the employed

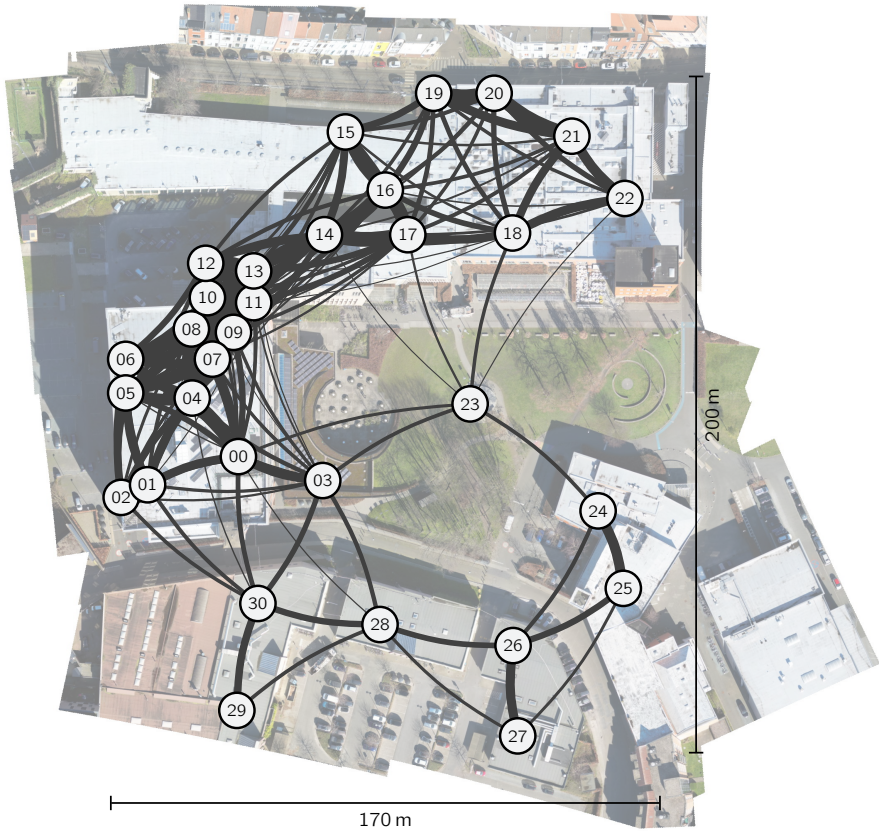


Figure 5.17: Overview of the placement of the IoT nodes in the university campus deployment.

prolonged sampling mechanism (see Section 5.3.1). This is indeed the case. The results in Figure 5.18 show that, for values of T_{measure} below 6.5 min, the PDR decreases significantly as T_{measure} decreases further. When T_{measure} is set to 3 min, for example, nodes with three child nodes experience a drop in PDR to only 52%. It is worth noting that this simulation takes the energy optimization w.r.t. the preamble duration into account (as discussed in Section 5.3.1): preamble duration is adjusted throughout the graph to the optimal value according to T_{measure}. Evidently, energy consumption increases with an increased network throughput as more and longer messages are sent in the network.

2. **Aggregation time.** It is well-established that extending the time that nodes wait for incoming messages to aggregate (i.e., increasing the aggregation

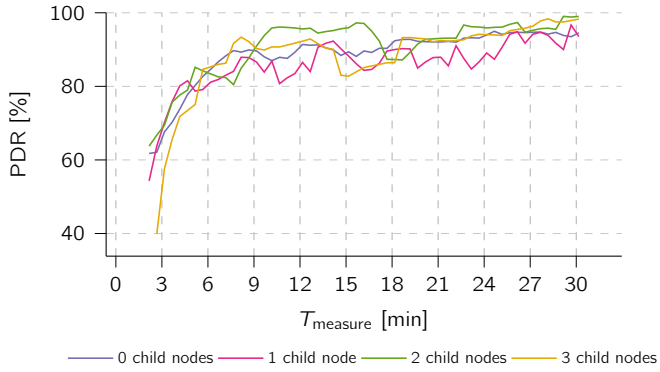


Figure 5.18: Simulated packet delivery ratio (PDR) across the multi-hop network plotted w.r.t. the measurement interval T_{measure}.

timer T_a) can benefit aggregation and energy efficiency. Figures 5.19 and 5.20 show the relationship between T_a and $\alpha_{aggregation}$ and energy efficiency, respectively, for nodes throughout the network.

Figure 5.19 demonstrates that $\alpha_{aggregation}$ increases with an increase in T_a. However, nodes with a larger number of child nodes (two or more, in this example) experience a more gradual increase in $\alpha_{aggregation}$ because the likelihood of data being forwarded from one of their child nodes increases with the number of child nodes. The aggregation ratio stabilizes for all nodes at around 90 % when $T_a/T_{\text{measure}} = 0.5$.

The energy per byte spent on transmitting data ($E_{\text{TX},B}$) is closely linked to the aggregation ratio $\alpha_{\text{aggregation}}$ as more aggregation causes $E_{\text{TX},B}$ to decrease. Yet, $E_{\text{TX},B}$ inherently also takes the size and amount of the aggregated data into account. As shown in Figure 5.20, as more message aggregation takes place, less energy per byte is required. It is worth noting that the potential energy savings increase when a node has more child nodes, as more data can be aggregated. $E_{\text{TX},B}$ stabilizes for all forwarding nodes at $T_a/T_{\text{measure}} = 0.5$: saving 69 % in energy consumption. By maximizing T_a, nodes with more than two child nodes can save up to 75 % of transmission energy, yet at the expense of a disproportionate extra latency. However, increasing T_a also results in an increasingly higher latency, which is disproportionate to the energy savings already obtained at $T_a/T_{\text{measure}} = 0.5$ (69 %).

It's worth noting that simulations also demonstrate how leaf nodes can save additional energy. By aggregating messages, fewer messages are transmitted, which minimizes overhearing by leaf nodes. As a result, less

energy is consumed during the receive process, leading to a possible energy reduction of up to 11 %.

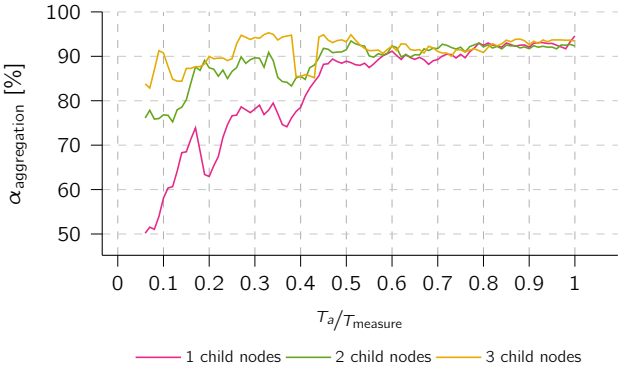


Figure 5.19: Simulated aggregation ratio $\alpha_{\text{aggregation}}$ across the multi-hop network w.r.t. the ratio of the aggregation timer value T_a to T_{measure}.

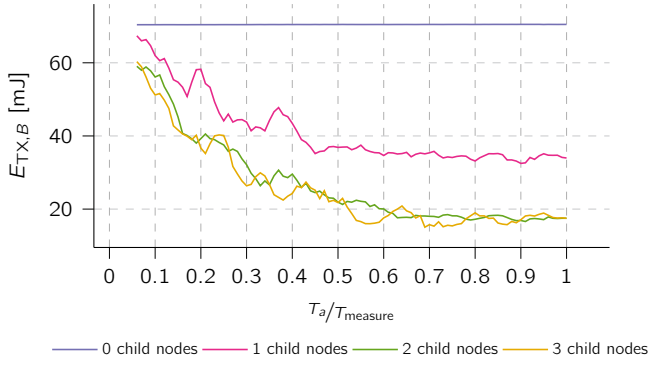


Figure 5.20: Simulated energy per byte spent by transmitting data $E_{\text{TX},B}$ across the multi-hop network w.r.t. the ratio of the aggregation timer value T_a to T_{measure}. It is clear that, for forwarding nodes, when T_a increases, the energy $E_{\text{TX},B}$ decreases significantly until $T_a/T_{\text{measure}} = 0.5$, after which it stabilizes at a minimum.

3. **Randomization parameters.**[9] To prevent messages from colliding with those transmitted by nearby nodes, some network timers use randomized values for each cycle. For example, the time between two consecutive CAD cycles and the aggregation timer includes a randomization margin.

[9] The results for these simulations are published in the open source repository of the multi-hop simulator: github.com/DRAMCO/LoRa-multihop-simulator

Simulations have shown that introducing randomization, as opposed to having timers set to zero, is crucial for preventing synchronized transmission among neighboring nodes. This synchronization occurs due to the nodes overhearing packets and subsequently aligning their timers based on these receptions. By introducing randomization, collisions are reduced, leading to improved PDR and decreased energy consumption per successfully transferred bit. However, simulations have shown that randomization need only extend up to a small margin around the original timer value. Beyond this margin, additional randomization does not yield further benefits and does not significantly enhance network performance in terms of latency or energy consumption.

4. **Sensor data size.** Increasing the data size in the multi-hop network results in higher energy consumption for all nodes since this increases the airtime required to transmit the data (see Figure 5.21). In addition, simulations have shown that when the data size is increased to, for example, 64 bytes, the aggregation ratio $\alpha_{\text{aggregation}}$ can decrease by up to 9 % (see Figure 5.22). This is because the aggregation buffers fill up faster, reducing the amount of aggregation that can take place and leading to a further slight increase in energy consumption. Despite the prolonged airtime, the network's PDR is not significantly affected by increasing the data size, according to simulations.

5. **LoRa spreading factor (SF) and transmit power.** By increasing the employed Spreading Factor (SF) or raising the transmit power, a node gains the ability to reach more distant nodes, effectively skipping intermediary hops. This hop-skipping approach has the potential to reduce overall energy consumption. However, it's essential to consider that increasing output power or SF also raises transmission energy. Furthermore, an elevated SF impacts the duration of a single CAD, which is based on the duration of a single LoRa symbol. This trade-off between raising SF or transmit power and the network's overall energy consumption becomes evident for the campus experiment shown in Figure 5.23. In this network, the sum of the energy consumption for all nodes is plotted across various output power levels and SFs. The average number of hops across the network is also plotted. Notably, the most energy-efficient configuration for this network is SF 7 at the lowest transmit power setting. Herein, no energy can be saved by increasing the SF or transmit power. However, this is because of the limited number of hops within this network, where the average hop count reaches a maximum of only 1.1 hops. To explore a scenario with more hops, consider a network consisting of a single line of ten nodes placed in succession. The results are depicted in Figure 5.24. In this case, it becomes evident that a higher output power leads to a reduction in total

energy consumption, reaching a minimum at a transmit power of 14 dBm when utilizing SF 7-9. Notably, for SF 10-12, energy consumption remains relatively stable across varying transmit power values, as the message hop count has already been minimized by using these large SFs. This illustrates that while increasing the transmit power or SF reduces the number of hops, it does not necessarily guarantee a reduction in energy consumption. The degree of potential energy savings largely relates to the specific geographical layout of the network. In networks with an increased number of hops, there is greater potential for energy savings. The simulator presented can aid in estimating the optimal configuration for a given network.

The assessment presented above provide valuable insights into the configuration of the deployed multi-hop network. Based on these insights, we determine the optimal network parameters for deployment, as summarized in Table 5.4. For general IoT monitoring use cases, where fast update rates are not required, we have established a measurement interval of $T_{\text{measure}} = 30\,\text{min}$ to conserve energy. To balance energy efficiency and latency, the aggregation timer ($T_{a,\text{max}}$) has been set to half of the measurement interval, i.e., 15 min. To minimize network latency, the minimum aggregation timer ($T_{a,\text{min}}$) has been set to 0 min. The size of sensor measurements has been fixed at 12 B, as previously set in Section 5.2.2.

5.6.2.2 Real-World Multi-Hop Performance

In order to evaluate the performance of the multi-hop platform, we deployed 30 sensor nodes and one gateway across the university campus (see Figure 5.17) using the "Dramco Uno" platform as described in Section 5.4. Each sensor node was configured to transmit an incrementing counter, enabling us to assess criteria such as the PDR for any given node. The data received at the gateway node (node *00*) was logged for subsequent analysis. The network was configured as listed in Table 5.4. The experiments were conducted in a 48-hour window.

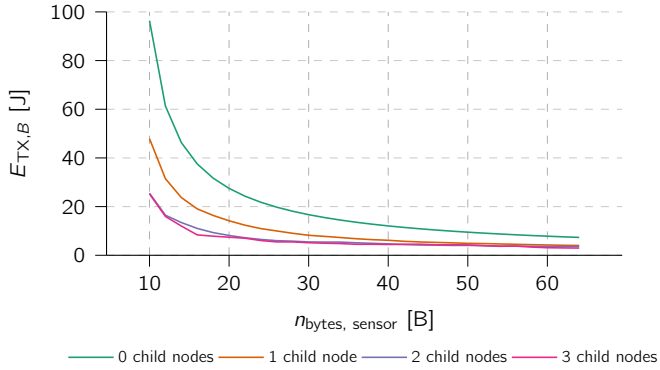


Figure 5.21: Simulated transmit energy per byte $E_{TX,B}$ across the multi-hop network w.r.t. the data size of one sensor value ($n_{bytes,\ sensor}$).

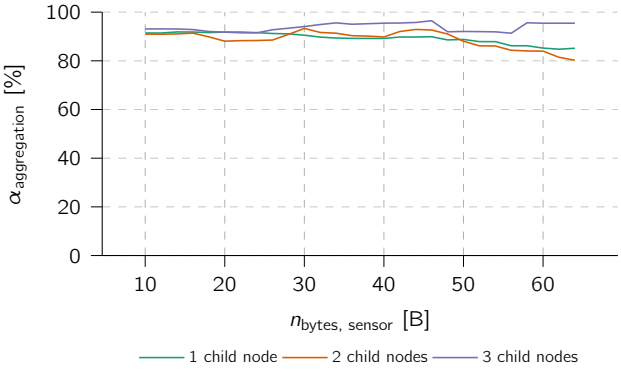


Figure 5.22: Simulated aggregation ratio $\alpha_{aggregation}$ across the multi-hop network w.r.t. the data size of one sensor value ($n_{bytes,\ sensor}$).

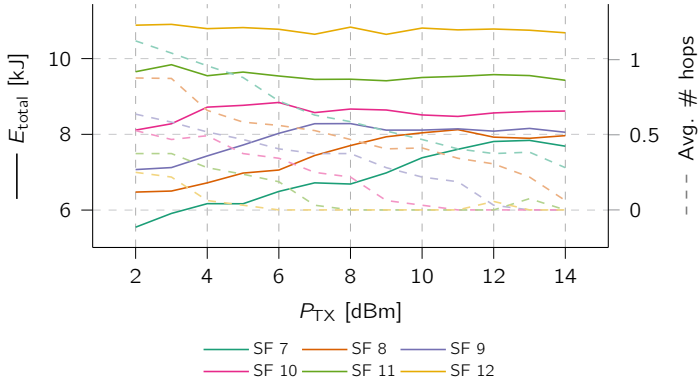


Figure 5.23: The collective energy (E_{total}) consumed in a day within the deployed network, w.r.t. both the transmit power and the selection of SF. The dashed lines indicate the average number of hops from the central gateway for different the configurations.

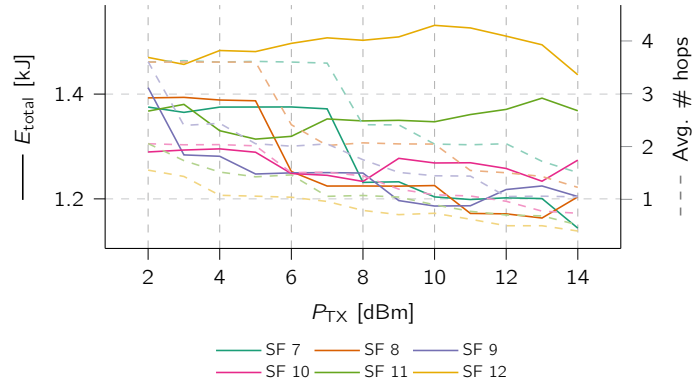


Figure 5.24: The collective energy (E_{total}) consumed in a day w.r.t. to both the transmit power and the selection of SF. The simulated network is a network with 10 nodes placed on a single line and spaced 35 m apart in an urban scenario. The dashed lines indicate the average number of hops from the central gateway for different the configurations.

Table 5.4: Network parameters used for the network, experimentally deployed on the university campus.

Parameter	Symbol	Value
Route discovery	T_{route}	6 h
Measure interval	T_{measure}	30 min
Minimal aggregation timer	$T_{a,\text{min}}$	0 s
Initial aggregation timer	$T_{a,0}$	12.5 min
Maximal aggregation timer	$T_{a,\text{max}}$	15 min
Aggregation upstep	$T_{a,\text{upstep}}$	1 min
Aggregation downstep	$T_{a,\text{downstep}}$	30 s
TX buffer size		150 B
Payload size		12 B
LoRa spreading factor	SF	7
LoRa bandwidth	BW	500 kHz
Preamble width	T_{preamble}	1.91 s

The first step taken by the network is self-configuration, i.e., determining the optimal route for message transmission. This process creates a tree-structured network as shown in Figure 5.25.

The majority of the network's nodes are situated at a distance of zero or one hop from the gateway. Three nodes are located at two hops from the gateway, and another three are at three hops.

The gateway sends out a route discovery message every 6 hours (as defined by T_{route} in Table 5.3). Therefore, the routes for each node may change every 6 hours due to varying SNR values across links (see Section 5.3.3). In Figure 5.25, the path taken by the majority of messages is represented in black, while alternative paths, taken by the minority of messages, are colored grey.

It should be noted that for some links, the trade-off between adding multiple hops with a good SNR and adding (multiple) extra hops to the communication is not straightforward for some nodes: these switch between two options, highly based on SNR_I. For example, node *18* is able to communicate directly with the gateway or via nodes *17* and *23*, which adds two extra hops to the path. Interestingly, node *18* sends out 68 % of messages via the extra hops, and only 32 % of messages directly to the gateway *00*, even though the PDR for both paths is the same (88 %). This is attributed to the dynamic nature of the environment in which these measurements were conducted, resulting in a fluctuating SNR and varying channel conditions, even in cases of a statically deployed network. The dynamic environment can introduce fluctuations in interference, multipath propagation, or shadowing. These factors can temporarily affect the SNR of a specific path between network hops, leading to fluctuations in the routing algorithm as it adapts

to these changes. In highly dynamic scenarios, it is recommended to reduce the route discovery interval (T_{route}) within the algorithm. For instance, modifying T_{route} to 2 h can help maintain a high PDR in such dynamic conditions, yet at the cost of battery autonomy in the network.

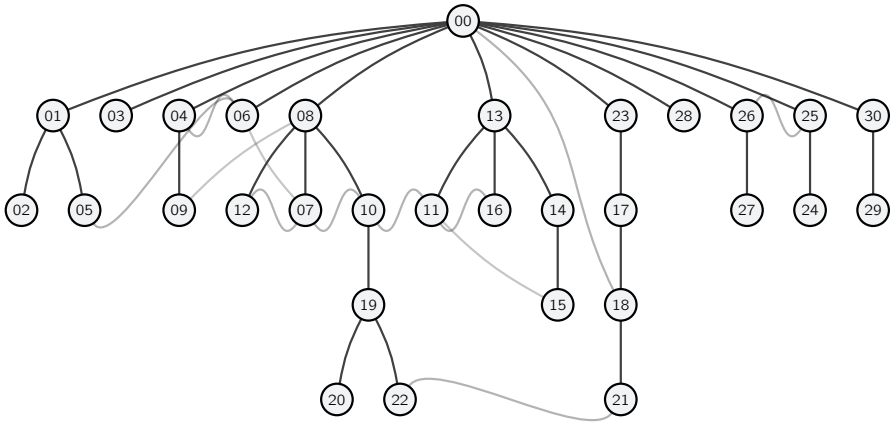


Figure 5.25: Tree diagram depicting the multi-hop routes established by the route discovery algorithm, based on experimental data. The diagram shows the routes of the majority of the messages, with alternative routes indicated in grey that were taken by a minority of messages. In this test, the route discovery mechanism is updated 8 times. Node positions correspond to those reported in Figure 5.17.

During the 48-hour logging period, the network's PDR was monitored for each node. The cumulative distribution function (CDF) for the PDR is shown in Figure 5.26. The results indicate that 90 % of the nodes achieved a PDR of at least 70 %. However, when analyzing the PDR in relation to the node's distance from the gateway, in terms of number of hops, it is clear that nodes located farther away from the gateway suffer from a lower PDR. The decrease in PDR observed in this multi-hop network with an aggregation scheme is due to its inherent limitations. In any multi-hop network, the average PDR of a message is the product of the average PDR values of all traversed links:

$$\text{PDR} = \prod_{l}^{L} \text{PDR}_l, \tag{5.8}$$

with PDR_l the PDR at link l. Additionally, since our aggregation scheme combines multiple sensor data measurements into a single message, a single lost message due to collision or other reasons can result in substantial data loss.

Furthermore, when assessing the PDR of a link, the potential presence of hidden nodes during message forwarding should also be considered. Hidden nodes in wireless networks refer to nodes unable to directly detect each other but communicating with a common receiver, such as a forwarding node or access point. When these hidden nodes simultaneously transmit data, their signals collide at the receiver, resulting in interference and potential data loss. Implementing a Request-to-Send (RTS) mechanism can mitigate data loss due to hidden nodes by effectively coordinating data traffic between nodes. However, this coordination comes at the expense of increased energy consumption.

The effectiveness of the aggregation mechanism is also evaluated in the experimental test setup. The aggregation ratio $\alpha_{\text{aggregation}}$ is determined for each node. The results are depicted in Figure 5.27: the CDF of the aggregation ratio $\alpha_{\text{aggregation}}$, with respect to the number of forwarding child nodes. The results show the effectiveness of the aggregation mechanism with the dynamic aggregation timer (as discussed in Section 5.3.5). Nodes with no child nodes, evidently do not forward any data. The chance that messages are received within the aggregation time, increases with the amount of forwarding child nodes. Yet, as the aggregation timer maximum $T_{a,\text{max}}$ is set at $T_{\text{measure}}/2$ (as determined in Section 5.6.2.1), some nodes still transmit messages that do not contain aggregated data. This is especially the case for nodes with one or two child nodes. Nodes that forward data from three child nodes, have an aggregation ratio of at least 90 %, effectively improving the energy per byte, as evaluated in Section 5.6.1 and Table 5.3.

5.7 Conclusion

This chapter presented a low-power and reliable multi-hop LoRa network aimed at providing IoT coverage in remote environments by sharing a multi-RAT connection across the network. In this multi-hop network, all data is assembled at a central gateway node. This node is equipped with both a LoRaWAN and NB-IoT network. Messages that arrive at the central gateway node get aggregated into a NB-IoT message, thereby forwarding sensor data of nodes in the network to the cloud. Multiple sensor data points of multiple nodes are aggregated into a single NB-IoT message, thereby lowering the energy consumption and latency as opposed to equipping all nodes with an extra NB-IoT connection. Furthermore, both subscription cost and hardware cost are reduced by only providing the gateway with a NB-IoT connection.

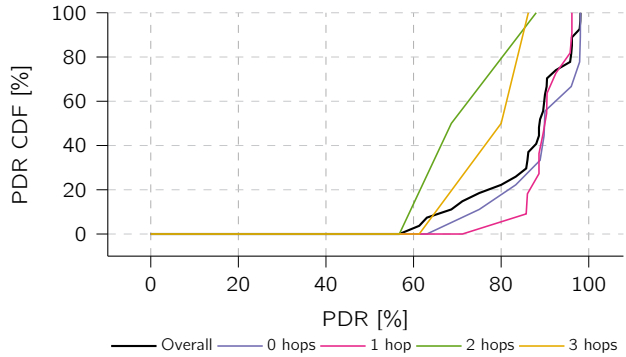


Figure 5.26: Cumulative distribution function (CDF) of the experimental packet delivery ratio (PDR) with respect to the node's position in the network (Figure 5.25). Nodes that require two or three hops tend to have a lower PDR compared to other nodes. When messages are aggregated with other sensor data, the loss of a single message may result in a substantial drop in PDR.

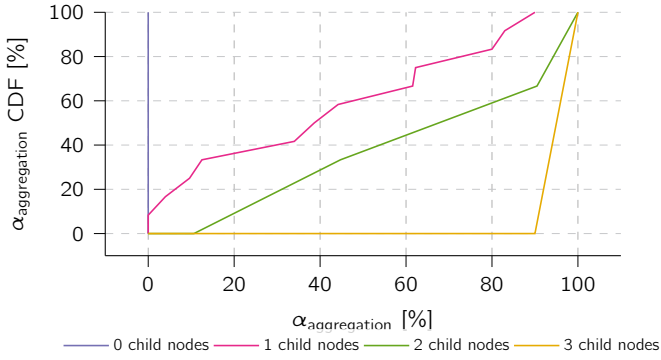


Figure 5.27: The Cumulative distribution function (CDF) of the aggregation ratio ($\alpha_{aggregation}$) with respect to the number of forwarding children and their positions in the network (see Figure 5.25). Nodes with three child nodes show a 90 % aggregation ratio, resulting in significant improvements in energy per byte.

By combining and optimizing prolonged preamble transmission and periodic CAD, a node's lifespan is ensured. For example, when nodes transmit data only every 6 hours, the lifespan can be increased from up to 2 days to up to 4 years. A cross-platform and open-source network protocol is proposed, which further reduces energy consumption by smartly routing and aggregating messages in between hops. A simulator framework to assess the effects of network configurations is presented. The experimental validation of this multi-hop protocol demonstrates its effectiveness in achieving energy gains and ensuring reliability. Initial practical tests reveal significant reductions in TX energy consumption, up to 61 %, by implementing the dynamic aggregation mechanism discussed earlier. Furthermore, the reliability of communication within the network is thoroughly evaluated through an extensive practical test setup. The results affirm that 90 % of the nodes achieve a PDR of at least 70 %, confirming the robustness and dependability of the presented network.

Chapter 6

Use Case Driven Adoption of LPWAN-based IoT

In this chapter, multiple case studies are analyzed and evaluated that exemplify the adoption of LPWAN technologies, whether single-RAT or multi-RAT, for a diverse set of IoT scenarios. The examination of these real-world applications is intended to reveal the benefits, challenges, and best practices associated with the IoT technologies presented in this doctoral study.

Throughout the chapter, the use cases presented in Chapter 1 are explored in depth and implemented in real-life prototypes. Each case study covers the design of low power IoT nodes, while explicitly focussing on what LPWAN technology is best suited. Herein lies a comprehensive presentation of the numerous opportunities for valorization of the IoT design techniques presented in this PhD study.

Part of the content presented in this chapter is based on:

- G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "Tracking Rental Bikes in Smart Cities: a Multi-RAT Approach," Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux, 2022. [LCVD22]
- B. Thoen[†], G. Callebaut[†], G. Leenders and S. Wielandt, "A Deployable LPWAN Platform for Low-Cost and Energy-Constrained IoT Applications," MDPI Sensors, vol. 19, no. 3: 585, January 2019, doi: 10.3390/s19030585. [TCLW19]
- S. Crul, G. Leenders and L. Van der Perre, "Glue-and-Play Sensing Solution for Remotely Monitoring Drinking Frequency of Horses," IEEE Sensors Applications Symposium, 2019, doi: 10.1109/SENSORS43011.2019.8956605. [CLV19]

Contributions: The author took a leading role in designing the hardware and firmware for the mentioned papers. In [LCVD22], the author conducted the experimental work and contributed to the paper's writing. The author made significant contributions to the experimental work and writing in [TCLW19] and [CLV19].

6.1 Single-RAT Case Studies

6.1.1 Tree Health Monitoring

Remote IoT sensors can play a vital role in monitoring the health of individual trees, allowing foresters to detect signs of stress or disease that may not be apparent to the naked eye. We consider the health of trees in the IoTree project [TCLW19].[1] In this project, custom-made low-power sensor nodes are deployed to monitor the health of trees (Figure 6.3). The IoTree nodes measure the temperature of the tree trunk combined with the outside temperature using two temperature sensors. From these two temperature measurements, an estimate of the stem juice flow can be derived. The on-board accelerometer is employed to monitor the inclination of the tree, monitoring large structural deformations of the tree, e.g., when the tree is blown down during a storm. The battery voltage is measured using the internal reference of the microcontroller.

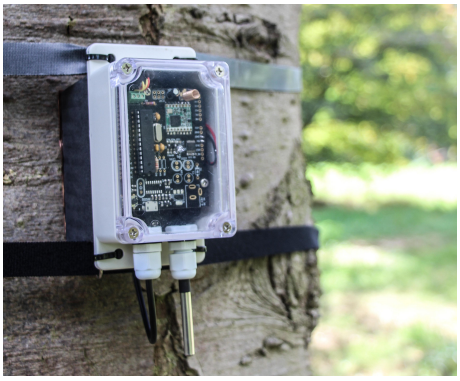


Figure 6.1: The IoTree node employed as a sensor for monitoring the health of trees.

[1]This project is also considered as a STEM project and science outreach for students, and consequently also focuses on the development of a versatile IoT node, within the applicable tree health monitoring use case.

The measured data is transmitted over the radio in byte efficient format, resulting in a data-frame of 6 bytes. Sensor data is collected every 15 minutes. The sensor data is not considered critical, thus a certain degree of packet loss can be tolerated. Latency is also not critical, up to a point of 2 hours: sensor data can arrive at the cloud 2 hours after taking the measurements. Network coverage is considered available for all three LPWANs: LoRaWAN, Sigfox, and NB-IoT, as these nodes are meant to be deployed in urban parks and forests.

For this project, a single-RAT solution is chosen due to the static nature of the use case, where the nodes remain in fixed positions without requiring variations in the requirements of the employed IoT technology. Additionally, all the considered IoT networks offer coverage, incorporating a multi-RAT enabled network would not yield any significant advantages.[2]

Compared to Sigfox and NB-IoT, LoRaWAN is especially chosen due to the superior energy efficiency when transmitting small payloads as demonstrated in Chapter 4. Additionally, the cost-effectiveness of LoRaWAN justifies its selection. By making use of the crowd-sourced The Things Network (TTN), connectivity is provided for free and additional gateways can be placed wherever needed. In this particular use case, the requirement for high QoS is not critical, allowing LoRaWAN to meet the project's needs.

The sensor makes use of the low-power design guidelines detailed in Chapter 2. In this use case, latency of up to 2 hours can be tolerated, making aggregating data possible. This can drastically prolong a node's autonomy, as discussed in Chapter 3. By aggregating data, a total of 48 byte is generated every 2 hours.

The obtained autonomy of the presented IoTree sensor node can be estimated using the calculator tool described in Section 3.1.3.1. An energy profile is obtained in which the readout of the sensors consumes an average of 65.7 mW for 190 ms. When transmitting on SF10, an autonomy of 7.8 years is estimated, when transmitting in less favorable network conditions (i.e., switching to continuous use of SF 12 via ADR), an autonomy of 4.9 years is projected (Using a battery pack consisting of two non-rechargeable AA batteries with a combined capacity of 2500 mAh).

[2]When alarm signals are needed to be transmitted when a tree leans too much during a storm, an expansion to a multi-RAT approach would be beneficial to offer a higher QoS for these events. However, this consideration is not taken into account in this context.

6.1.2 Monitoring the Drinking frequency of Horses

The average horse consumes between 15.5 to 75.5 liters of water per day [Hin78]. Monitoring the drinking frequency of horses is crucial as a decrease in frequency can indicate various health issues, including the life-threatening symptom known as colic [BTdAF13]. Horse owners require a solution to track drinking frequency even when their horses are in remote locations such as riding schools, meadows, or during training sessions. While multiple sensor products for horses exist, such as [Ari19], these are unsuitable for remote monitoring and lack sufficient battery life for a set-and-forget implementation.

In some cases, stables are in rural or remote areas. Easy retrofitting of the sensor-system needs to be possible and compatible with the variety of already installed drinking bowls. A flexible capacitive sensing solution to be glued in the actual drinking bowl is conceived, as depicted in Figure 6.2. Conceptually, it measures a change in capacity when the water level in the bowl changes, as well as when the horse touches the bowl.

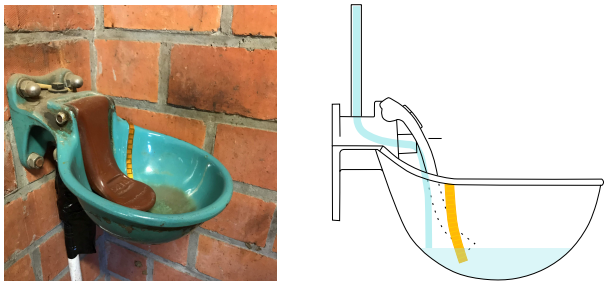


Figure 6.2: Depiction of the drinking sensor glued inside a drinking bowl.

As mains power is often absent, we aim for a battery powered implementation. An energy model is constructed to accurately predict the autonomy of this sensing device.

Sensor readings occur frequently with a 30-second interval, but the actual readings are not time-sensitive and do not require frequent transmission. In the context of long-term evolution monitoring, data aggregation is possible, preferably at least once every hour. Each sensor reading consumes 2 bytes of payload. [3]

[3] In this context, data is not processed on the node itself to generate useful water consumption data, which would result in additional energy savings. However, a careful evaluation of the sensor's functionality must be conducted prior to implementing such an approach.

Considering these requirements, two suitable IoT technologies are identified: NB-IoT and LoRaWAN. Due to the static nature of this use case, a single-RAT solution is chosen. In this context, LoRaWAN is preferred due to its better true cost of ownership. LoRaWAN nodes can be connected without recurring subscription costs through a free and crowd-sourced LoRaWAN network: The Things Network (TTN). Sensor deployment is limited to specific locations such as fields or barns, allowing easy extension of network coverage by adding additional LoRaWAN gateways nearby. In our setup, the gateway is placed in the main building of the farm, while the sensor nodes are deployed in nearby stables or outdoor drinking posts.

The gathered data is sent to a cloud application where the drinking frequency of the horse can be monitored.

In LoRaWAN, battery autonomy can be optimized by combining 111 measurements per LoRaWAN packet (2 bytes per measurement for total of 222 bytes in a maximum payload). Using the autonomy calculator described in Section 3.1.3.1, we project a worst case battery-powered autonomy of 14 months (acknowledgement in second RX-window, transmit power 14 dB and a spreading factor of 12) using two AA batteries with a capacity of 2500 mAh each. Best case, an autonomy of nearly 11 years can be expected (acknowledgement in first RX-window, transmit power 2 dB and spreading factor 7).

6.2 Multi-RAT Devices: Case Studies

6.2.1 Sound Pollution Monitoring

Noise pollution is a prevalent issue in urban environments, as highlighted by studies [VVV+10]. However, current understanding of environmental sounds is limited and primarily relies on simulated noise maps derived from large-scale models [10]. These noise maps, which are not frequently updated and can take several years to be revised, offer only a rough estimation of long-term average noise levels, lacking real-time accuracy.

Real-time acoustic monitoring systems present a solution by offering up-to-date insights into sound pollution and instant detection of acoustic events. By incorporating such systems, responsible authorities can be automatically alerted and informed promptly. As a result, the monitoring system serves as an impartial source of information, complementing human complaints and ensuring a comprehensive understanding of the noise pollution landscape.

The monitoring of noise pollution in smart city landscapes is considered by the Dramco research group in the IOASE project [BTC+19]. In this project, a custom-made low-power sensor node equipped with a microphone was developed to monitor the level of noise pollution. The microphone serves as a calibrated noise level sensor.

Figure 6.3: The IOASE node employed as a sound level sensor in an urban park and city square.

In accordance with the "think before you talk" principle described in Chapter 2, the sensor node has been designed to record and transmit data only when absolutely necessary. To achieve this, a specific microphone with a wake-on-sound mode has been selected. This mode allows the microphone to wake up from sleep when a predefined sound pressure level is exceeded. By employing this feature, system-level power consumption is reduced to less than 18 µW during inactive periods. When the sound level surpasses the predetermined threshold, a 2-byte payload is sent with minimal latency. To prevent constant activation of the sound level sensor, a cooldown period of 10 minutes is initiated following each transmitted sound event.

For this original prototype, a single-RAT solution based on LoRaWAN was chosen due to its energy efficiency when dealing with small payloads, as only sound level values needed to be transmitted. One limitation of the approach employed in the IOASE project is that it solely transmits sound levels, providing no insight into the nature or origin of the intrusive sounds. To address this, we have expanded the IOASE prototype to incorporate the recording, processing, and potentially transmission of sound signals associated with loud noise events. This enhancement enables the classification of these sound events in the cloud, allowing for a comprehensive understanding of the disruptive noises. By analyzing these classified sound events, appropriate actions can be taken to mitigate the disturbances and create a quieter environment.

In this more complex use-case, regular sound level measurements are taken at intervals of every 10 minutes to establish a baseline value for the surrounding sound level of the node. Additionally, a threshold-based recording mechanism is implemented, wherein a 1-second recording is triggered upon detection of loud sound. This recording is intended to facilitate classification and identification of the source of the loud noise in the cloud [IDK+21; MK19]. With a sampling rate of 8 kHz and a bit depth of 8 bits, a recording of raw audio results in a payload size of 8 kB.[4]

Considering the need to transmit two distinct payload sizes within this use case, we encounter two messages that need to be send: (1) A periodic transmission of small sensor data, which serves to relay the sensor's status, such as sound levels, to the cloud at regular intervals. (2) Occasional transmission of larger 1-second sound samples, triggered only sporadically, when a sound disturbance is detected. To address these divergent data transmission requirements, a multi-RAT strategy is implemented. For the periodic, smaller sensor data updates, LoRaWAN is selected as the preferred communication technology, showcasing superior energy efficiency for these specific payload sizes. In contrast, the transmission of larger, sporadic data originating from sound recordings relies on NB-IoT to optimize energy efficiency when transmitting these larger payloads. This multi-RAT approach preserves energy compared to exclusive reliance on either NB-IoT or LoRaWAN for all data transmissions. A comprehensive analysis of the advantages of combining LoRaWAN and NB-IoT can be found in Section 4.3. Moreover, this strategy capitalizes on the lower latency inherent in NB-IoT for the transmission of larger payloads in contrast to LoRaWAN.

The multi-RAT platform as presented in Section 4.2.1 is used for dynamic switching between LoRaWAN and NB-IoT. A detailed energy profile is recorded to calculate the obtained node autonomy (see Table 6.1).

The autonomy of the sound monitoring IoT node strongly depends on the number of additional sound fragments that are transmitted, whether multi-RAT or single-RAT. This is depicted in Figure 6.4. A node that transmits two sound recordings per day has a projected autonomy of 2.2 years when using the multi-RAT scheme,

[4] Alternatively, instead of sending raw audio signals as payload, the IoT node could calculate a spectogram of the audio signal, which constitutes in performing multiple fast Fourier transform (FFT) calculations and the subsequent transmission of the most prominent frequency components. While this reduces the amount of data to be transmitted, the increased energy consumption due to calculating the spectogram should also be considered [ELX22; MLC23]. Other methods to reduce the amount of transmitted data generally amount to lossy data compressing techniques (e.g., adopting the MP3 codec), yet these would require intensive compute power of the IoT node, which is generally a low-cost device, and would reduce classification accuracy [MLC23]. Consequently, these methods are not put in practice in this study.

Table 6.1: Energy consumption profile of the developped sound monitoring platform.

Event	Energy [mJ]
Sound level measurement	0.6
Sound recording (1 s)	74
Sending LoRaWAN 12 byte	55.8
Sending LoRaWAN 51 byte	112
Sending NB-IoT 12 byte (avg.)	1,916
Sending NB-IoT 8 kB (avg.)	10,760

which is an increase with a factor of 2.6 when compared to sending all messages over NB-IoT or an increase of 54 % when compared to a LoRaWAN-only setup. The respective gains of employing a multi-RAT setup with respect to a single-RAT setup are depicted in Figure 6.5. When implementing the LoRaWAN single-RAT approach, an important drawback is the added latency of at least 1 hour to transmit the larger sound recording, which makes the sensor's usage as an alarm less suitable.

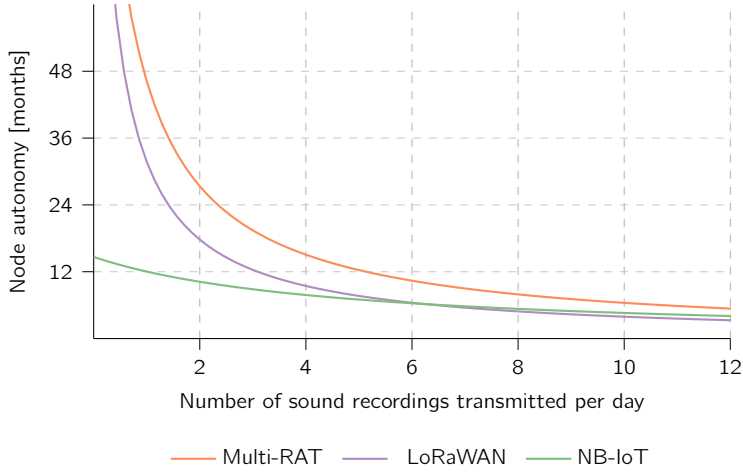


Figure 6.4: The projected node autonomy with respect to the number of sound recordings transmitted for a multi-RAT setup and two single-RAT setups. Transmission over LoRaWAN is assumed to employ SF 10 and NB-IoT is assumed to use CE level 0.

In this particular use-case, it becomes evident that a multi-RAT approach offers significant advantages in terms of energy efficiency, primarily due to the diversity of payload sizes. Although the potential increase in node hardware size might be a concern, it holds less significance since ultra-small node sizes are not a requirement

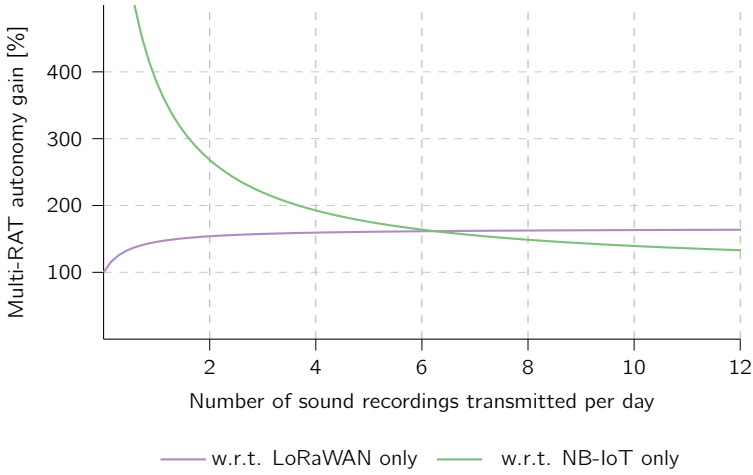


Figure 6.5: Autonomy gained by employing a multi-RAT setup on the IoT node with respect to the number of sound recordings. Transmission over LoRaWAN is assumed to employ SF 10 and NB-IoT is assumed to use CE level 0.

for this use-case. Nevertheless, it is crucial to consider the additional hardware and subscription costs, as they should be factored into the project budget.

6.2.2 Track and Trace: Rental bikes

As cities ban polluting cars from city centers, bikes are becoming ever more popular for traveling last-mile, short distances in a commute towards and inside the city center. The concept of bike-sharing involves a collection of bikes, spread out across a city. For customers of the bike sharing platform, these bikes are freely available for rent - when not in use. Two main problems, afflicting bike sharing concepts, are the loss of or theft of bikes [ZZDB15] and the limited energy budget for tracking bikes. Traditionally, these bikes are being tracked using a GNSS connection for localization, and a cellular 2G connection for cloud connection. However, both technologies consume large amounts of energy, which is troublesome for battery-operated trackers. To reduce power consumption, Croce *et al.* [CGG+19] studied the applicability of LoRaWAN cloud communication for rental bikes.

The research conducted in Chapter 4 highlights several areas with non-existent or highly unreliable coverage for LoRaWAN. Underground parking lots proof to be the most challenging environment for tracking bikes. Installing LoRaWAN gateways

Figure 6.6: Bike lock that tracks rental bikes (on which the author advised during the initial design phase regarding the multi-RAT capabilities).

throughout an entire city to achieve comprehensive coverage is not a feasible solution. Opting for NB-IoT as the single-RAT solution would result in a lower node autonomy compared to the autonomy achieved by LoRaWAN. We propose a multi-RAT approach using both unlicensed (LoRaWAN) and licensed (NB-IoT) IoT technologies as both feature distinct advantages regarding QoS, cost and energy consumption. As demonstrated in Chapter 4, NB-IoT features higher QoS levels (including coverage, latency performance and payload size). However, LoRaWAN excels in both energy efficiency for small payloads and cost efficiency. Combining both NB-IoT and LoRaWAN in a single IoT node, enables dynamic switching between these IoT technologies depending on various circumstances. When combining multiple wireless IoT technologies for the purpose of coverage enhancement, the resulting service area is effectively the total of all individual service areas combined.

As bikes are often left behind in underground parking lots, coverage in these challenging environments is of paramount importance. We studied coverage levels of both LoRaWAN and NB-IoT in multiple underground parking lots in the city of Bruges (Belgium). Our findings are summarized in Table 6.2. These indicate the superior performance of NB-IoT due to the variety in CE level, resulting in a larger MCL due to the used modulation schemes and packet repetitions. By boosting CE levels, transmit power and packet repetitions are increased.

Through the integration of both LoRaWAN and NB-IoT, significant energy reduction is achieved by prioritizing LoRaWAN connectivity when available. The use of confirmed messages allows the IoT node to discern the presence of LoRaWAN coverage. When LoRaWAN coverage is unavailable, the IoT node transitions to

Table 6.2: Coverage performance comparison of LoRaWAN and NB-IoT in underground parking lots. Both RSSI and RSRP are signal level and quality indicators.

Level	LoRaWAN	NB-IoT
0	✓ (SF 12, -113 dBm RSSI)	✓ (CE 0, -61 dBm RSRP)
-1	✗	✓ (CE 2, -127 dBm RSRP)
-2	✗	✓ (CE 2, -118 dBm RSRP)
-3	✗	✓ (CE 1, -120 dBm RSRP)
-4	✗	✓ (CE 2, -133 dBm RSRP)
-5	✗	✓ (CE 2, -134 dBm RSRP)

using NB-IoT communication to ensure reliable data delivery. This solution is especially valuable in challenging coverage scenarios, such as underground parking lots, where NB-IoT leverages its implementation of CE levels to provide robust connectivity. Thus providing a drastically improved geographical service area than only using LoRaWAN. Strategically deploying cost-effective LoRaWAN gateways at locations with suboptimal LoRaWAN coverage further enhances the overall network reach, while prolonging the battery autonomy of the tracking nodes. For instance, consider the scenario of transmitting GNSS location data with a 6 B payload to attain 3 m accuracy. In contexts where LoRaWAN is accessible, the energy consumption for transmitting such a payload is notably reduced in comparison to exclusive reliance on NB-IoT. Detailed insights into this energy-saving aspect are provided in Section 4.3, where it's demonstrated that utilizing LoRaWAN results in energy savings of up to 73 % for small payload sizes when compared to utilizing NB-IoT.

To further minimize the energy consumption of the localization node, the deployed LPWAN technologies could also be leveraged to estimate the location of the rental bike [LBS+21; RJN+17]. This approach could lead to additional energy savings by reducing reliance on the power-intensive GNSS localization.[5]

6.3 Multi-RAT Supported Networks: Wildfire Monitoring Case Study

The increasing frequency and severity of wildfires pose a significant threat to ecosystems, property, and human lives. In recent years, there has been a growing

[5]The localization opportunities through multi-RAT LPWAN systems are acknowledged in this PhD as promising for employing a multi-RAT solution, however are not studied in depth and provide further research opportunities.

interest in leveraging the possibilities of IoT networks to detect wildfires at their early stages. By deploying an extensive network of connected sensors and devices, it becomes possible to detect and respond to wildfire outbreaks swiftly, enabling timely intervention and mitigation measures [Som11; MT22; DS05].

To monitor the development of wildfires, sensors can be deployed at various points in the forest. These sensors need to periodically transmit their reading. If a sensor detects abnormal conditions indicative of a potential wildfire, such as a sudden increase in temperature or high levels of smoke, it needs to promptly send an alert to the underlying framework.

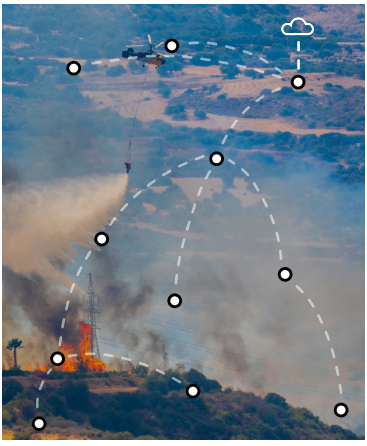


Figure 6.7: Concept of how sensor data for wildfire detection can be assembled at a central low-power gateway node, using the presented multi-hop, multi-RAT supported network.

While there may be partial cellular or IoT network coverage at the edges of a remote forest, achieving comprehensive coverage across the entire forest area that requires monitoring is not realistic or feasible. The dense vegetation and challenging terrain within forests can obstruct signals, hindering reliable network coverage deep within the forest. To address this challenge, a multi-RAT enabled network solution, as explained in Chapter 5, can be employed. This solution leverages the multi-hop topology of multi-RAT, allowing for deep coverage penetration, while also utilizing NB-IoT nodes at the forest edges to provide IoT network support.

For the specific case of wildfire risk evaluation and early detection, essential sensor readings such as temperature, humidity, light intensity, smoke, and soil moisture data need to be transmitted [Som11]. Considering these data points fit in a total payload of 5 bytes per reading, these measurements can be aggregated and sent in 6-hour intervals, resulting in a 30-byte payload. Hence, the assumptions

regarding the proposed multi-hop network of low data throughput and non-latency-dependent requirements are valid in this scenario. Additionally, the nodes need to continuously monitor these sensor readings for thresholds and provide alarm functionality in case of urgent detection.

Based on simplified calculations similar to those shown in Figure 5.7, we estimate a maximum battery autonomy of 3.9 years for a battery pack consisting of two AA batteries (2500 mAh). However, in the proposed multi-hop network, the energy consumption of the nodes depend largely on data throughput from other sensor nodes and node placement. An accurate projection of battery autonomy requires careful consideration of these variables, which can be simulated using the multi-hop simulation framework, presented in Section 5.5.

6.4 Conclusion

In this section, the applicability of both single-RAT and multi-RAT IoT technologies is demonstrated through comprehensive use case studies. For predominantly static and non-dynamic scenarios like the presented tree health monitoring and horse drinking monitoring, a single-RAT solution is generally preferred. However, for more advanced and dynamic use cases, the application can greatly benefit from some form of multi-RAT connectivity. For instance, combining LoRaWAN and NB-IoT can save energy by leveraging their different energy efficiency optimizations for payload diversity, as illustrated in our in-depth analysis of sound pollution monitoring. Increased coverage or improved QoS can also drive the adoption of a multi-RAT setup, as seen in the connectivity requirements for tracking rental bikes. Furthermore, in large deployments that necessitate high coverage and energy efficiency for monitoring or alarm purposes, the proposed multi-RAT enabled multi-hop network proves to be well-suited, as demonstrated in the tracking of wildfire risk.

Chapter 7

Conclusions and Outlook

At the end of each chapter, a specific conclusion was provided. In this final chapter, a more general conclusion is presented along with an exploration of potential future approaches to enhance energy efficiency of IoT.

To encourage future research, all hardware, firmware, measurement data, and processing files have been made available as open-source resources. The corresponding chapters contain the necessary links for accessing these materials.

7.1 Conclusions

The contributions and conclusions of this study are summarized and evaluated in reference to the research questions posed in the introduction of this PhD study (Chapter 1).

How can low-power, low-cost and long-range capable IoT devices be designed?
Design guidelines for developing low-power, low-cost, and long-range capable IoT devices are provided in Chapter 2. It is shown that a comprehensive approach, encompassing hardware component selection, suitable communication technology choices, and tailored firmware development for specific use cases, is essential.

A detailed analysis of an IoT node's energy consumption reveals two primary challenges: minimizing idle power consumption and reducing energy usage in wireless communication.

To address idle energy consumption, two strategies are proposed: "sleep as much as possible" and "race to sleep". The former emphasizes avoiding unnecessary hardware activation and maximizing sleep intervals for all components, while the latter focuses on reducing high-power state durations through the utilization of dedicated hardware features.

A thorough examination of both the PHY and MAC layers is conducted to explore how LPWAN technologies improve energy efficiency while enabling long-distance communication. Additionally, a two-fold strategy, referred to as the "think before you talk" approach, is suggested. This involves validating or reducing the amount of transmitted data and aggregating non-time critical data into a single message.

Furthermore, a novel approach is proposed to further lower energy expenditure in wireless communication: implementing a multi-RAT setup of complementary LPWAN technologies (Chapter 4, Chapter 5). The objective is to lower the energy expenditure of the communication while meeting the specific requirements of the use case regarding data throughput or latency.

Can the energy expenditure of IoT LPWAN technologies be characterized?
To optimize energy efficiency, it is essential to begin with a fine-grained energy and behavior model that allows for comprehensive analysis and trade-off assessments. In this study, the analysis is conducted for three specific technologies: LoRaWAN, Sigfox, and NB-IoT, grounded in empirical data. The identification of key factors influencing energy consumption plays a crucial role in this analysis.

For LoRaWAN and Sigfox, it becomes evident that the payload size significantly impacts the energy consumed per packet. Larger payloads result in a reduced energy consumption per byte, emphasizing the importance of data aggregation in a single packet rather than transmitting each data point separately. Additionally, parameters such as the employed SF and transmit power in LoRaWAN strongly influence a node's autonomy. However, these parameters can be controlled and optimized through the LoRaWAN ADR algorithm. To accurately determine node autonomy, an energy calculator tool has been developed for both LoRaWAN and Sigfox. This tool incorporates not only the energy expenditure related to wireless communication but also accounts for sleep power consumption and battery self-discharge.

Energy consumption in NB-IoT is more difficult to project due to the cellular nature

of the network and its deployment dependent operation. In this study, energy consumption is evaluated in practice, considering the various stages involved in NB-IoT communication. The measurements conducted highlight the increased variance of energy consumption in a low-SNR regime. Furthermore, the findings emphasize that the energy distribution among different communication states (e.g., join, transmit) heavily depends on the network quality.

Can the combination of implementing multiple LPWAN technologies on a single IoT device be exploited to support new applications and further increase energy efficiency?
The implementation of multiple LPWAN technologies on a single IoT device can indeed be leveraged to support new applications and further enhance energy efficiency. In order to investigate the energy benefits of a multi-RAT system on IoT nodes, a dedicated multi-RAT platform was developed, incorporating LoRaWAN, Sigfox, and NB-IoT technologies. This platform enables the measurement of energy consumption on a per-packet basis for each IoT technology, while also considering relevant network and contextual parameters (e.g., PDR and node location).

The findings presented in Chapter 4 demonstrate that energy expenditure of IoT nodes can be reduced through the implementation of complementary LPWAN technologies. Specifically, the combination of LoRaWAN and NB-IoT proves to be well-suited for decreasing energy consumption and supporting more complex and dynamic use cases.

For instance, in scenarios where an application requires both small and larger message payloads, the choice of IoT technology significantly affects energy consumption and latency. By incorporating a fallback strategy, if the most energy-efficient technology fails to transmit a message successfully, communication critical events can be addressed. The benefits of implementing multiple LPWAN technologies include energy-efficient operation for variable payload sizes, ensuring timely delivery for latency-critical messages, extending coverage, and improving quality of service. Notably, for applications that involve variable payload sizes, energy consumption can be reduced by a factor of 5, when compared to using NB-IoT alone, and a 33 % reduction can be achieved when using LoRaWAN alone. These findings are applied to specific use cases in Chapter 6: sound monitoring in smart cities and the track and trace of rental bikes.

Can a multi-hop network be leveraged to share multiple LPWAN technologies across an IoT network to increase network coverage and reduce the energy consumption?
The integration of multiple LPWAN technologies in a single IoT network can improve both energy consumption of a node and extend the overall geographical range of the network. To address the energy consumption challenges typically associated with central, gateway-like devices, a novel network is proposed in Chapter 5, which relies on prolonged preamble sampling and uses a multi-hop topology.

In this approach, a central node with multi-RAT capabilities acts as a gateway and supports both LoRaWAN and NB-IoT connections. Instead of continuously listening for incoming packages, which consumes substantial energy, the network utilizes an asynchronous preamble sampling based technique. Network-wide energy consumption is further reduced by smartly routing across the network and aggregating messages between hops.

Comparing the preamble-based technique to continuous reception windows, the results demonstrate a notable improvement in node autonomy. With the implemented approach, nodes experience increased battery life, lasting up to four years compared to the previous two days. Moreover, by leveraging the aggregation mechanism, a reduction of up to 61 % in TX energy consumption is observed.

The effectiveness of the multi-hop network is validated through simulations and experimental testing. The experiments confirm the reliability of the multi-hop technique, with 90 % of nodes achieving a Packet Delivery Ratio (PDR) of at least 70 % when transmitting data points to the central IoT gateway.

7.2 Outlook

The multi-RAT approaches presented in this PhD research offer effective means to enhance the energy efficiency and feature set of IoT nodes.

In the specific context of the multi-RAT on-device strategies discussed in Chapter 4, they have been implemented to address distinct use cases. These strategies require customization to align with the specific requirements of each use case. To dynamically select the most suitable candidate, an algorithm can be devised. Such algorithms must consider the application, context, and wireless technologies generalized requirements and features. Machine learning techniques are envisioned to be crucial in achieving a fully autonomous and responsive algorithm. However, it

is important to assess whether the benefits gained from these algorithms outweigh the additional resource demands they impose.

The proposed multi-RAT capable network, as outlined in Chapter 5, demonstrates the feasibility of creating an energy-efficient multi-RAT gateway-like device that also functions as a remote and energy-constrained IoT node. Future research could explore other mechanisms to determine the optimal routes between multiple neighbors, for example RL-MAC [LE06], which employs reinforcement learning to optimize data throughput and energy consumption in WSNs. Furthermore, optimizing network load across multiple nodes is not discussed in this implementation of the multi-hop protocol. By incorporating both the remaining available energy and the current communication load of an IoT node in the route discovery mechanism, a more even distribution of energy consumption across the network can be obtained. In order to improve the reliability of the network, acknowledgments and re-transmissions might be employed, positively affecting the PDR. Moreover, a data broadcast mechanism might be implemented in order to quickly relay time-critical messages such as alarms from nodes to the gateway. Finally, it is important to highlight that the current network design is well suited for static scenarios characterized by large-scale, low-density networks. The potential for deploying this network in highly dynamic settings, such as deployment on mobile vehicles (public transport) or in extremely dense network environments, warrants further investigation. These contexts inherently demand distinct approaches, including reevaluating elements like the utilization of asynchronous communication.

In a more general sense, we propose three possible research tracks to further reduce the environmental impact of IoT devices beyond energy consumption: (i) energy harvesting for smaller batteries, (ii) lowering the impact of network deployments, and (iii) towards circular IoT.

Energy harvesting for smaller batteries. In this PhD research, our primary focus was on reducing the energy consumption of individual IoT nodes. However, prolonging the autonomy of an IoT node can also be achieved through an alternative approach: increasing the available energy to the node. This can be accomplished by either increasing the battery capacity or better yet, adopting rechargeable batteries combined with energy harvesting techniques.

Energy harvesting offers a means to supply energy to devices during deployment, potentially extending their autonomy to an unlimited duration and reducing the need for large batteries in IoT nodes. There are various sources of energy that can be harvested, either from the ambient environment or external sources. Ambient

sources refer to energy that is naturally present in the surroundings, while external sources involve intentionally delivering energy to the devices [VLC+22].

Recent advancements in energy harvesting solutions have specifically targeted small IoT devices [MAG+17; LS15]. While solar or light radiation in general is the most commonly utilized source for energy harvesting in many use cases, other sources with lower energy yield can also proof their worth. These include RF energy, vibration energy, and thermal energy.

Lowering the impact of network deployments. In this PhD research, the focus is primarily on the energy consumption of individual IoT nodes. To achieve a comprehensive understanding of the environmental impact of IoT deployments, it is necessary to assess the energy consumption and environmental implications of not only the IoT nodes but also the supporting infrastructure. This includes gateways, cellular base stations, and other network components.

For instance, the deployment of NB-IoT alongside existing cellular 4G networks offers the advantage of utilizing the existing infrastructure, thereby reducing the environmental impact associated with building new infrastructure. However, it is crucial to consider the energy consequence of implementing NB-IoT on base stations. Recent research has indicated that integrating NB-IoT on base stations can lead to a considerable increase in their baseline energy consumption due to NB-IoT signaling. In this scenario, the base station is impeded from entering sleep mode, resulting in higher energy consumption.

Towards circular IoT. In the context of implementing IoT, increasing the energy efficiency of IoT nodes has been recognized as a crucial means to mitigate its environmental impact, as highlighted in Chapter 1. However, it is also essential to consider the environmental implications associated with the resulting electronic waste (e-waste) [Hig18].

The deployment of IoT technologies contributes to a significant carbon footprint due to the utilization of scarce raw materials and energy consumption throughout the manufacturing, operation, and recycling processes. To address these concerns, the concept of Green IoT has emerged as a novel research area aimed at reducing the carbon footprint associated with IoT [FLF21].

Green IoT requires a paradigm shift in architecture and raises pertinent research questions concerning design and operational procedures. Herein, the goal should be to develop a novel architectural approach that enables future IoT devices to

adopt a fully circular lifecycle, encompassing sustainable design practices and autonomous recovery mechanisms.

By embracing a circular approach, IoT devices can be designed with considerations for resource efficiency, recyclability, and reusability [ZLSN15]. This entails utilizing environmentally friendly materials, optimizing energy consumption during device operation, and implementing effective recycling and recovery strategies at the end of their lifecycle.

Datasheets, Reference Manuals and Software

[1] "AN0025: Peripheral Reflex System (PRS)", Silicon Labs Laboratories, Tech. Rep., Rev. 1.08, online: www.silabs.com/documents/public/application-notes/an0025-efm32-prs.pdf [accessed: December 2020].

[2] "BME680 Low Power Gas, Pressure, Temperature & Humidity Sensor", Bosch Sensortec, Tech. Rep., Rev. 1.5, online: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme680-ds001.pdf [accessed: December 2020].

[3] "Efficient Super-Capacitor Charging with TPS62740", Texas Instruments, Tech. Rep. SLVA678, p. 10.

[4] "eXtreme Low-Power (XLP) PIC Microcontrollers: An Introduction to Microchip's Low Power Devices, AN1267", Tech. Rep.

[5] "Internet of Things in Logistics", Cisco, DHL, Tech. Rep., online: www.dhl.com/discover/content/dam/dhl/downloads/interim/full/dhl-trend-report-internet-of-things.pdf [accessed: April 2023].

[6] *LoRaWAN Regional Parameters*, RP2-1.0.3. [Online]. Available: https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/.

[7] "Narrowband Internet of Things Measurements: Application Note", Rhode & Schwarz, Tech. Rep. [Online]. Available: https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_application/application_notes/1ma296/1MA296_2e_NB-IoT_Measurements.pdf (visited on 04/21/2023).

[8] "Narrowband Internet of Things Whitepaper", Rohde & Schwarz, Tech. Rep., p. 42. [Online]. Available: cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ma266/1MA266_0e_NB_IoT.pdf.

[9] "NB LTE – Battery lifetime evaluation", 3GPP, Tech. Rep. RP-151393, p. 42.

[10] *Noise in Europe 2014*, EEA Report No 10/2014. [Online]. Available: https://www.eea.europa.eu/publications/noise-in-europe-2014.

[11] *Reducing Downlink Current Drain of Sigfox Devices by Using Discontinuous RX*, DLK-OPT-AN 1.1. [Online]. Available: https://support.sigfox.com/docs/downlink-optimization.

[12] *Short Range Devices (SRD) operating in the frequency range 25 MHz to 1 000 MHz; Part 2: Harmonised Standard for access to radio spectrum for non specific radio equipment*, ETSI EN 300 220-2 V3.2.1. [Online]. Available: https://www.etsi.org/deliver/etsi_en/300200_300299/30022002/03.02.01_60/en_30022002v030201p.pdf.

[13] *SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver*. [Online]. Available: https://www.semtech.com/uploads/documents/DS_SX1276-7-8-9_W_APP_V5.pdf.

[14] "Whitepaper: Making Sense of Industrial IoT Platforms", Everactive, Tech. Rep., online: https://everactive.com/whitepapers/emerging-industrial-internet-technology-can-reduce-costs-and-dangers-in-steam-systems/ [accessed: April 2023].

[15] "PIC Microcontroller Low Power Tips 'n Tricks, DS01146B", Microchip Technology Inc., Tech. Rep., 2009.

[16] "Cellular System Support for Ultra-Low Complexity and Low Throughput Internet of Things (CIoT)", 3GPP, Tech. Rep., Sep. 2015, p. 4. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2719.

[17] *LoRa™ Modulation Basics*, AN1200.22, Semtech, 2015. [Online]. Available: https://www.semtech.com/uploads/documents/an1200.22.pdf.

[18] *Alkaline Manganese Dioxide Handbook and Application Manual*, English, version 800-383-7323, Energizer, 2018, 15 pp., October 2, 2019.

[19] "NB-IoT Deployment Guide to Basic Feature Set Requirements, Version 2.0", *White paper, Official Document CLP*, vol. 28, 2018.

[20] *Discovery kit for LoRaWAN, Sigfox, and LPWAN protocols with STM32L0*, version 4.0, [Accessed on: November 19, 2021], Jun. 2019. [Online]. Available: st.com/resource/en/data_brief/b-l072z-lrwan1.pdf.

[21] "IoT Connections Forecast: The Rise of Enterprise", GSMA, Tech. Rep., 2019, online: www.gsma.com/iot/resources/iot-connections-forecast-the-rise-of-enterprise/ [accessed: April 2023].

[22] "Narrowband IoT Delivers Insights from the Largest NB-IoT Indoor Measurement Campaign", Deutsche Telekom AG, Tech. Rep., Jan. 2019, p. 4.

[23] "TPS63900 1.8-V to 5.5-V, 75-nA IQ Buck-boost Converter with Input Current Limit and DVS", Texas Instruments, Tech. Rep. RSLVSET3D, 2020, p. 32.

[24] "State of IoT -— Spring 2022", IOT Analytics, Tech. Rep., 2022, online: www.iot-analytics.com/product/state-of-iot-spring-2022/ [accessed: April 2023].

[25] Atmel, *Peripheral Power Consumption in Standby Mode for SAM D Devices, AT11491*, English, Atmel, Jul. 2015.

[26] T. S. Chew and M. Tuttle, "Application Note 2 – Using the VM1010 Wake-on-Sound Microphone and Zero Power Listening™Technology", Vesper Technologies, Tech. Rep., 2018, Rev. 4, online: www.digikey.be/nl/pdf/v/vesper/using-vm1010-wake-on-sound-microphone [accessed: December 2020].

[27] G. Leenders, *NB-IoT Power Measurements*, version 1.0, Nov. 29, 2019. DOI: 10.5281/zenodo.3557239. [Online]. Available: http://dx.doi.org/10.5281/zenodo.3557239.

[28] G. Leenders, *Multi RAT for IoT - Source files*, version 1.0, [Accessed on: April 20, 2021], Apr. 14, 2021. DOI: 10.5281/zenodo.4687297. [Online]. Available: http://dx.doi.org/10.5281/zenodo.4687297.

[29] M. Lesund, *LTE-M vs NB-IoT Field Test: How Distance Affects Power Consumption*. [Online]. Available: https://devzone.nordicsemi.com/nordic/nordic-blog/b/blog/posts/ltem-vs-nbiot-field-test-how-distance-affects-power-consumption.

[30] Microchip, *PIC24FJ128GA310 Family, DS30009996G*, English, Microchip, Mar. 2014.

[31] Microchip, *PIC16(L)F1717/8/9: Cost-Effective 8-Bit Intelligent Analog Flash Microcontrollers, DS40001740C*, English, Microchip, Sep. 2018.

[32] N. Sornin, A.Yegin, et al., *LoRaWAN™ 1.1 Specification*, https://lora-alliance.org/sites/default/files/2018-04/lorawantm_specification_-v1.1.pdf, version 1.1, 2017.

[33] G. Ottoy, G. Leenders, and G. Callebaut, *LoRaWAN EFM32*, https://github.com/DRAMCO/LoRaWAN_EFM32, doi: 10.5281/zenodo.1209414. DOI: 10.5281/zenodo.1209414. [Online]. Available: https://github.com/DRAMCO/LoRaWAN_EFM32.

[34] M. Salas, "Low Power Design Basics: How to Choose the Optimal Low Power MCU for your Embedded System", Silicon Labs Laboratories, Tech. Rep.

[35] Silicon Labs, "AN0007.0: MCU and Wireless MCU Energy Energy Modes", 2017, Rev. 1.10. [Online]. Available: https://www.silabs.com/documents/public/application-notes/an0007.0-efm32-ezr32-series-0-energymodes.pdf.

[36] N. Sornin, M. Luis, T. Eirich, T. Kramp, and O. Hersent, "LoRaWAN Specification", *LoRa Alliance*, 2015.

[37] J. C. Zuniga and B. Ponsard, "Sigfox System Description", *LPWAN IETF97, Nov. 14th*, vol. 25, 2016.

Bibliography

[AAP+17] P. Andres-Maldonado, P. Ameigeiras, J. Prados-Garzon, J. Navarro-Ortiz, and J. M. Lopez-Soler, "Narrowband IoT Data Transmission Procedures for Massive Machine-Type Communications", *IEEE Network*, vol. 31, no. 6, pp. 8–15, 2017.

[ABBB18] T. Adame Vázquez, S. Barrachina-Muñoz, B. Bellalta, and A. Bel, "HARE: Supporting Efficient Uplink Multi-Hop Communications in Self-Organizing LPWANs", *Sensors*, vol. 18, no. 1, p. 115, 2018.

[ADA+19] K. A. Aldahdouh, K. A. Darabkh, W. Al-Sit, *et al.*, "A survey of 5G emerging wireless technologies featuring LoRaWAN, Sigfox, NB-IoT and LTE-M", in *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, IEEE, 2019, pp. 561–566.

[ADM+21] B. Al Homssi, K. Dakic, S. Maselli, H. Wolf, S. Kandeepan, and A. Al-Hourani, "IoT Network Design Using Open-source LoRa Coverage Emulator", *IEEE Access*, vol. 9, pp. 53 636–53 646, 2021.

[AFL+19] A. Abrardo, A. Fort, E. Landi, M. Mugnaini, E. Panzardi, and A. Pozzebon, "Black Powder Flow Monitoring in Pipelines by Means of Multi-Hop LoRa Networks", in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4. 0&IoT)*, IEEE, 2019, pp. 312–316.

[AGD+11] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. A. Imran, D. Sabella, M. J. Gonzalez, O. Blume, and A. Fehske, "How Much Energy is Needed to Run a Wireless Network?", *IEEE Wireless Communications*, vol. 18, no. 5, pp. 40–49, 2011. DOI: 10.1109/MWC.2011.6056691.

[AGZ21] A. Andreadis, G. Giambene, and R. Zambon, "Monitoring Illegal Tree Cutting through Ultra-Low-Power Smart IoT Devices", *Sensors*, vol. 21, no. 22, p. 7593, 2021.

[AKA+19] M. S. Aslam, A. Khan, A. Atif, S. A. Hassan, A. Mahmood, H. K. Qureshi, and M. Gidlund, "Exploring Multi-Hop LoRa for Green Smart Cities", *IEEE Network*, vol. 34, no. 2, pp. 225–231, 2019.

[AMH20] S. Aslam, M. P. Michaelides, and H. Herodotou, "Internet of Ships: a Survey on Architectures, Emerging Applications, and Challenges", *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9714–9727, 2020.

[AMSP18] A. Azari, G. Miao, C. Stefanovic, and P. Popovski, "Latency-Energy Tradeoff Based on Channel Scheduling and Repetitions in NB-IoT Systems", in *2018 IEEE Global Communications Conference*, IEEE, 2018, pp. 1–7.

[AMT+20] D. Abruzzese, A. Micheletti, A. Tiero, M. Cosentino, D. Forconi, G. Grizzi, G. Scarano, S. Vuth, and P. Abiuso, "IoT Sensors for Modern Structural Health Monitoring. A New Frontier", *Procedia Structural Integrity*, vol. 25, pp. 378–385, 2020.

[And+19] P. Andrés-Maldonado *et al.*, "NB-IoT M2M Communications in 5G Cellular Networks", 2019.

[AP19] A. Abrardo and A. Pozzebon, "A Multi-Hop LoRa Linear Sensor Network for the Monitoring of Underground Environments: the Case of the Medieval Aqueducts in Siena, Italy", *Sensors*, vol. 19, no. 2, p. 402, 2019.

[Ari19] Arioneo. "Orscana". (2019), [Online]. Available: `https://www.arioneo.com/en/home/care/` (visited on 05/13/2019).

[AVT+16] F. Adelantado, X. Vilajosana, P. Tuset-Peiró, B. Martínez, and J. Melià, "Understanding the Limits of LoRaWAN", *CoRR*, vol. abs/1607.08011, 2016. arXiv: `1607.08011`. [Online]. Available: `http://arxiv.org/abs/1607.08011`.

[AWW05] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: a Survey", *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.

[BDWL10] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung, "MAC Essentials for Wireless Sensor Networks", *IEEE Communications Surveys Tutorials*, vol. 12, no. 2, pp. 222–248, 2010. DOI: `10.1109/SURV.2010.020510.00058`.

[BKD+22] K. Banti, I. Karampelia, T. Dimakis, A.-A. A. Boulogeorgos, T. Kyriakidis, and M. Louta, "LoRaWAN Communication Protocols: A Comprehensive Survey under an Energy Efficiency Perspective", in *Telecom*, MDPI, vol. 3, 2022, pp. 322–357.

[BMF+08] G. Berardinelli, L. A. Maestro, S. Frattasi, M. I. Rahman, and P. Mogensen, "OFDMA vs. SC-FDMA: Performance Comparison in Local Area IMT-A Scenarios", *IEEE Wireless Communications*, vol. 15, no. 5, pp. 64–72, 2008.

[BSD16] A. Boyden, V. K. Soo, and M. Doolan, "The Environmental Impacts of Recycling Portable Lithium-Ion Batteries", *Procedia CIRP*, vol. 48, pp. 188–193, 2016, The 23rd CIRP Conference on Life Cycle Engineering, ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2016.03.100`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S2212827116300701`.

[BSMP11] R. V. Biradar, S. Sawant, R. Mudholkar, and V. Patil, "Multihop Routing in Self-Organizing Wireless Sensor Networks", *International Journal of Computer Science Issues (IJCSI)*, vol. 8, no. 1, p. 155, 2011.

[BTC+19] C. Buyle, B. Thoen, B. Cox, M. Alleman, S. Wielandt, and L. De Strycker, "Ultra-Low-Power Smart Sensing Platform for Urban Sound Event Monitoring", in *Proceedings of the 2019 Symposium on Information Theory and Signal Processing in the Benelux*, WIC; http://www. wic. org/publications. html, 2019, pp. 35–40.

[BTdAF13] C. B. Belli, J. P. F. Távora, R. de Azevedo Ferreira, and W. R. Fernandes, "Evaluation of Equine Albumin Solution in Fluid Therapy in Horses with Colic", *Journal of Equine Veterinary Science*, vol. 33, no. 7, pp. 509–514, 2013.

[BVB+19] M. Bezunartea, R. Van Glabbeek, A. Braeken, J. Tiberghien, and K. Steenhaut, "Towards Energy Efficient LoRa Multihop Networks", in *2019 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, IEEE, 2019, pp. 1–3.

[CGB05] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-Constrained Modulation Optimization", *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2349–2360, 2005.

[CGG+19] D. Croce, D. Garlisi, F. Giuliano, A. L. Valvo, S. Mangione, and I. Tinnirello, "Performance of LoRa for Bike-Sharing Systems", in *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*, IEEE, 2019, pp. 1–6.

[CKA+21] G. Caso, K. Kousias, Ö. Alay, A. Brunstrom, and M. Neri, "Nb-iot random access: Data-driven analysis and ml-based enhancements", *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11 384–11 399, 2021.

[CKA13] L. W. Couch, M. Kulkarni, and U. S. Acharya, *Digital and Analog Communication Systems*. Citeseer, 2013, vol. 8.

[CLB+19] G. Callebaut, G. Leenders, C. Buyle, S. Crul, and L. V. der Perre, "LoRa Physical Layer Evaluation for Point-to-Point Links and Coverage Measurements in Diverse Environments", in *2019 European Conference on Networks and Communications (EuCNC)*, 2019.

[CLL+19] W.-L. Chen, Y.-B. Lin, Y.-W. Lin, R. Chen, J.-K. Liao, F.-L. Ng, Y.-Y. Chan, Y.-C. Liu, C.-C. Wang, C.-H. Chiu, *et al.*, "AgriTalk: IoT for Precision Soil Farming of Turmeric Cultivation", *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5209–5223, 2019.

[CLV+21] G. Callebaut, G. Leenders, J. Van Mulders, G. Ottoy, L. De Strycker, and L. Van der Perre, "The Art of Designing Remote IoT Devices — Technologies and Strategies for a Long Battery Life", *Sensors*, vol. 21, no. 3, 2021, ISSN: 1424-8220. DOI: 10.3390/s21030913. [Online]. Available: https://www.mdpi.com/1424-8220/21/3/913.

[CLV19] S. Crul, G. Leenders, and L. Van der Perre, "Glue-and-Play Sensing Solution for Remotely Monitoring Drinking Frequency of Horses", in *2019 IEEE SENSORS*, IEEE, 2019, pp. 1–4.

[CMVG17] L. Casals, B. Mir, R. Vidal, and C. Gomez, "Modeling the energy performance of LoRaWAN", *Sensors*, vol. 17, no. 10, p. 2364, 2017, ISSN: 1424-8220. DOI: 10.3390/s17102364. [Online]. Available: http://dx.doi.org/10.3390/s17102364.

[COG+22] J. Cappelle, G. Ottoy, S. Goossens, H. Deprez, J. Van Mulders, G. Leenders, G. Callebaut, and K. Leuven, "IoT with a Soft Touch: A Modular Remote Sensing Platform for STE(A)M Applications", in *2022 IEEE Sensors Applications Symposium (SAS)*, IEEE, 2022, pp. 1–6.

[COP20] G. Callebaut, G. Ottoy, and L. V. d. Perre, "Optimizing Transmission of IoT Nodes in Dynamic Environments", in *2020 International Conference on Omni-layer Intelligent Systems (COINS)*, 2020, pp. 1–5. DOI: 10.1109/COINS49042.2020.9191674.

[COv19] G. Callebaut, G. Ottoy, and L. van der Perre, "Cross-Layer Framework and Optimization for Efficient Use of the Energy Budget of IoT Nodes", in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2019, pp. 1–6. DOI: 10.1109/WCNC.2019.8885739.

[Cri06] S. C. Cripps, *RF Power Amplifiers for Wireless Communications*. Artech house Norwood, MA, 2006, vol. 2.

[CV20] G. Callebaut and L. Van der Perre, "Characterization of LoRa Point-to-Point Path Loss: Measurement Campaigns and Modeling Considering Censored Data", *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1910–1918, 2020. DOI: 10.1109/JIOT.2019.2953804.

[CVG+22] J. Cappelle, J. Van Mulders, S. Goossens, G. Leenders, and L. Van der Perre, "Strategies for Increasing Longevity of IoT Devices", in *Pre-Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux*, 2022, p. 36.

[DDH+20] E. Dons, S. De Craemer, H. Huyse, J. Vercauteren, D. Roet, F. Fierens, W. Lefebvre, C. Stroobants, and F. Meysman, "Measuring and Modelling Exposure to Air Pollution with Citizen Science: the CurieuzeNeuzen Project", in *ISEE Conference Abstracts*, vol. 2020, 2020.

[DG18] J. Dias and A. Grilo, "LoRaWAN Multi-Hop Uplink Extension", *Procedia Computer Science*, vol. 130, pp. 424–431, 2018.

[DGG+20] D. Daji, K. Ghule, S. Gagdani, A. Butala, P. Talele, and H. Kamat, "Cloud-Based Asset Monitoring and Predictive Maintenance in an Industrial IoT System", in *2020 International Conference for Emerging Technology (INCET)*, IEEE, 2020, pp. 1–5.

[dORRM19] F. C. de Oliveira, J. J. Rodrigues, R. A. Rabêlo, and S. Mumtaz, "Performance Delay Comparison in Random Access Procedure for NB-IoT, LoRa, and SigFox IoT Protocols", in *2019 IEEE 1st Sustainable Cities Latin America Conference (SCLA)*, IEEE, 2019, pp. 1–6.

[DR01] G. M. Djuknic and R. E. Richton, "Geolocation and Assisted GPS", *Computer*, vol. 34, no. 2, pp. 123–125, 2001.

[DS05] D. M. Doolin and N. Sitar, "Wireless Sensors for Wildfire Monitoring", in *Smart Structures and Materials 2005: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, SPIE, vol. 5765, 2005, pp. 477–484.

[EGR20] J. J. L. Escobar, F. Gil-Castiñeira, and R. P. D. Redondo, "JMAC Protocol: a Cross-Layer Multi-Hop Protocol for LoRa", *Sensors*, vol. 20, no. 23, p. 6893, 2020.

[ELX22] H. E. Erdem, H. Leung, and N. Xie, "Energy Neutral Urban Noise Monitoring and Classification With LoRaWAN Based IoT", in *2022 IEEE Sensors*, IEEE, 2022, pp. 1–4.

[Fer10] G. Ferrari, *Sensor Networks: Where Theory Meets Practice*. Springer Science & Business Media, 2010.

[FFL+15] A. C. Fischer, F. Forsberg, M. Lapisa, S. J. Bleiker, G. Stemme, N. Roxhed, and F. Niklaus, "Integrating MEMS and ICs", *Microsystems & Nanoengineering*, vol. 1, no. 1, May 2015, ISSN: 2055-7434. DOI: 10.1038/micronano.2015.5. [Online]. Available: http://dx.doi.org/10.1038/micronano.2015.5.

[FKYP21] A. Farhad, D.-H. Kim, J.-S. Yoon, and J.-Y. Pyun, "Feasibility Study of the LoRaWAN Blind Adaptive Data Rate", in *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN)*, IEEE, 2021, pp. 67–69.

[FLF21] P. Fraga-Lamas, S. I. Lopes, and T. M. Fernández-Caramés, "Green IoT and edge AI as Key Technological Enablers for a Sustainable Digital Transition Towards a Smart Circular Economy: an Industry 5.0 Use Case", *Sensors*, vol. 21, no. 17, p. 5745, 2021.

[Fra16] J. Fraden, *Handbook of Modern Sensors*, 5th ed. Springer, 2016.

[Fri46] H. T. Friis, "A Note on a Simple Transmission Formula", *Proceedings of the IRE*, vol. 34, no. 5, pp. 254–256, 1946. DOI: 10.1109/JRPROC.1946.234568.

[FS20] G. Fastellini and C. Schillaci, "Precision Farming and IoT Case Studies Across the World", in *Agricultural Internet of Things and Decision Support for Precision Smart Farming*, Elsevier, 2020, pp. 331–415.

[GPE+21] B. Geldhof, J. Pattyn, D. Eyland, S. Carpentier, and B. Van de Poel, "A Digital Sensor to Measure Real-Time Leaf Movements and Detect Abiotic Stress in Plants", *Plant Physiology*, vol. 187, no. 3, pp. 1131–1148, 2021.

[GVV+19] C. Gomez, J. C. Veras, R. Vidal, L. Casals, and J. Paradells, "A Sigfox Energy Consumption Model", *Sensors*, vol. 19, no. 3, p. 681, Feb. 2019, ISSN: 1424-8220. DOI: 10.3390/s19030681. [Online]. Available: http://dx.doi.org/10.3390/s19030681.

[GYI+22] M. GRARI, M. YANDOUZI, I. IDRISSI, M. BOUKABOUS, O. MOUSSAOUI, M. AZIZI, and M. MOUSSAOUI, "Using IoT and ML for Forest Fire Detection, Monitoring, and Prediction: a Literature Review", *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 19, 2022.

[HDMH18] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A Survey of LoRaWAN for IoT: from Technology to Application", *Sensors*, vol. 18, no. 11, p. 3995, 2018.

[Hig18] S. Higginbotham, "The Internet of Trash [Internet of Everything]", *IEEE Spectrum*, vol. 55, no. 6, pp. 17–17, 2018.

[Hin78] M. Hinton, "On the Watering of Horses: a Review", *Equine Veterinary Journal*, vol. 10, no. 1, pp. 27–31, 1978.

[HLL+17] A. Hoglund, X. Lin, O. Liberg, A. Behravan, E. A. Yavuz, M. Van Der Zee, Y. Sui, T. Tirronen, A. Ratilainen, and D. Eriksson, "Overview of 3GPP Release 14 Enhanced NB-IoT", *IEEE network*, vol. 31, no. 6, pp. 16–22, 2017.

[HPSA18] B. Huang, Z. Pan, X. Su, and L. An, "Recycling of Lithium-Ion Batteries: Recent Advances and Perspectives", *Journal of Power Sources*, vol. 399, pp. 274–286, 2018, ISSN: 0378-7753. DOI: https://doi.org/10.1016/j.jpowsour.2018.07.116. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378775318308498.

[HSN04] D. Helms, E. Schmidt, and W. Nebel, "Leakage in CMOS circuits–an introduction", in *International Workshop on Power and Timing Modeling, Optimization and Simulation*, Springer, 2004, pp. 17–35.

[IDK+21] A. Iliev, M. Dewli, M. Kalkan, P. Prakash Kudva, and R. Turkar, "Acoustic Event Detection and Sound Separation for security systems and IoT devices", in *Proceedings of the 22nd International Conference on Computer Systems and Technologies*, 2021, pp. 34–39.

[IRS18] D. Ismail, M. Rahman, and A. Saifullah, "Low-Power Wide-Area Networks: Opportunities, Challenges, and Directions", in *Proceedings of the Workshop Program of the 19th International Conference on Distributed Computing and Networking*, 2018, pp. 1–6.

[JD20] E. M. Jovanovska and D. Davcev, "No Pollution Smart City Sightseeing Based on WSN Monitoring System", in *2020 Sixth international Conference on Mobile and Secure Services (MobiSecServ)*, IEEE, 2020, pp. 1–6.

[JPC14] M. Jimenez, R. Palomera, and I. Couvertier, *Introduction to Embedded Systems: Using Microcontrollers and the MSP430*. Jan. 2014, pp. 1–648, ISBN: 978-1-4614-3142-8. DOI: 10.1007/978-1-4614-3143-5.

[Kam11] R. Kamal, *Microcontrollers: Architecture, Programming, Interfacing and System Design*. Pearson Education India, 2011.

[KHA20a] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges", *Sensors*, vol. 20, no. 18, p. 5044, 2020.

[KHA20b] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges", *Sensors*, vol. 20, no. 18, p. 5044, Sep. 2020, ISSN: 1424-8220. DOI: 10.3390/s20185044. [Online]. Available: http://dx.doi.org/10.3390/s20185044.

[KJD+17] P. Kalhara, V. Jayasinghearachchi, A. Dias, V. Ratnayake, C. Jayawardena, and N. Kuruwitaarachchi, "TreeSpirit: Illegal Logging Detection and Alerting System Using Audio Identification over an IoT Network", in *2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, IEEE, 2017, pp. 1–7.

[KML+17] I. Z. Kovács, P. Mogensen, M. Lauridsen, T. Jacobsen, K. Bakowski, P. Larsen, N. Mangalvedhe, and R. Ratasuk, "LTE IoT Link Budget and Coverage Performance in Practical Deployments", in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, IEEE, 2017, pp. 1–6.

[KW07] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons, 2007.

[LBCV19] G. Leenders, C. Buyle, G. Callebaut, and L. Van der Perre, "An Experimental Evaluation of Power Optimization in NB-IoT", in *2019 Symposium on Information Theory and Signal Processing in the Benelux, Date: 2019/05/28-2019/05/29, Location: Ghent*, WIC, 2019, pp. 61–61.

[LBP23] L. Leonardi, L. L. Bello, and G. Patti, "MRT-LoRa: a Multi-Hop Real-Time Communication Protocol for Industrial IoT Applications over LoRa Networks", *Computer Communications*, vol. 199, pp. 72–86, 2023.

[LBS+21] Y. Li, J. Barthelemy, S. Sun, P. Perez, and B. Moran, "Urban Vehicle Localization in Public LoRaWan Network", *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 10 283–10 294, 2021.

[LCC+17] Y. Li, X. Cheng, Y. Cao, D. Wang, and L. Yang, "Smart Choice for the Smart Grid: Narrowband Internet of Things (NB-IoT)", *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1505–1515, 2017.

[LCO+21] G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre, and L. De Strycker, "Multi-RAT for IoT: The Potential in Combining LoRaWAN and NB-IoT", *IEEE Communications Magazine*, vol. 59, no. 12, pp. 98–104, 2021.

[LCO+23] G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre, and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling for Remote Sensing", *Sensors*, vol. 23, no. 11, p. 4994, 2023.

[LCOV17] G. Leenders, G. Callebaut, G. Ottoy, and L. Van der Perre, "Versatile Module for Connecting Sensors via Low Power Wide Area Network (LPWAN)", in *PIMRC 2017*, 2017.

[LCVD20] G. Leenders, G. Callebaut, L. Van der Perre, and L. De Strycker, "An Experimental Evaluation of Energy Trade-Offs in Narrowband IoT", in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, 2020, pp. 1–5. DOI: 10.1109/WF-IoT48130.2020.9221010.

[LCVD22] G. Leenders, G. Callebaut, L. Van der Perre, and L. De Strycker, "Tracking Rental Bikes in Smart Cities: a Multi-RAT Approach", in *Pre-Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux*, 2022, p. 66.

[LCVD23] G. Leenders, G. Callebaut, L. Van der Perre, and L. De Strycker, "Multi-RAT IoT–What's to Gain? An Energy-Monitoring Platform", in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, IEEE, 2023, pp. 1–5.

[LE06] Z. Liu and I. Elhanany, "RL-MAC: a Reinforcement Learning Based MAC Protocol for Wireless Sensor Networks", *International Journal of Sensor Networks*, vol. 1, no. 3-4, pp. 117–124, 2006.

[LHI08] M. Lehtinen, A. Happonen, and J. Ikonen, "Accuracy and Time to First Fix Using Consumer-Grade GPS Receivers", in *2008 16th International Conference on Software, Telecommunications and Computer Networks*, IEEE, 2008, pp. 334–340.

[Liu18] L. Liu, "IoT and a Sustainable City", *Energy Procedia*, vol. 153, pp. 342–346, 2018.

[LKPC15] W. Lee, J. Koo, Y. Park, and S. Choi, "Transfer Time, Energy, and Quota-Aware Multi-RAT Operation Scheme in Smartphone", *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 307–317, 2015.

[LLW+11] Y. Li, J. Lopez, P. Wu, W. Hu, R. Wu, and D. Y. C. Lie, "A SiGe Envelope-Tracking Power Amplifier With an Integrated CMOS Envelope Modulator for Mobile WiMAX/3GPP LTE Transmitters", *IEEE Transactions on Microwave Theory and Techniques*, vol. 59, no. 10, pp. 2525–2536, 2011. DOI: 10.1109/TMTT.2011.2164550.

[LOC+23] G. Leenders, G. Ottoy, G. Callebaut, L. Van der Perre, and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling", in *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2023, pp. 1–6.

[LOC22] G. Leenders, G. Ottoy, and G. Callebaut, "Dramco Uno: A Low-Entry IoT Learning Platform for STE(A)M-Oriented Education", in *Pre-Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux*, 2022, p. 135.

[LPP19] A. Lavric, A. I. Petrariu, and V. Popa, "Long Range SigFox Communication Protocol Scalability Analysis Under Large-Scale, High-Density Conditions", *IEEE Access*, vol. 7, pp. 35 816–35 825, 2019. DOI: 10.1109/ACCESS.2019.2903157.

[LS15] X. Liu and E. Sánchez-Sinencio, "An 86% Efficiency 12 μW Self-Sustaining PV Energy Harvesting System With Hysteresis Regulation and Time-Domain MPPT for IOT Smart Nodes", *IEEE Journal of Solid-State Circuits*, vol. 50, no. 6, pp. 1424–1437, 2015. DOI: 10.1109/JSSC.2015.2418712.

[LSW+17] O. Liberg, M. Sundberg, E. Wang, J. Bergman, and J. Sachs, *Cellular Internet of Things: Technologies, Standards, and Performance*. Academic Press, 2017.

[MABV19] B. Martinez, F. Adelantado, A. Bartoli, and X. Vilajosana, "Exploring the Performance Boundaries of NB-IoT", *IEEE Internet of Things Journal*, 2019.

[MAC+20] F. Michelinakis, A. S. Al-Selwi, M. Capuzzo, A. Zanella, K. Mahmood, and A. Elmokashfi, "Dissecting Energy Consumption of NB-IoT Devices Empirically", *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 1224–1242, 2020.

[MAG+17] M. Magno, F. A. Aoudia, M. Gautier, O. Berder, and L. Benini, "WULoRa: An energy efficient IoT end-node for energy harvesting and heterogeneous communication", in *Design, Automation Test in Europe Conference Exhibition (DATE), 2017*, 2017, pp. 1528–1533. DOI: 10.23919/DATE.2017.7927233.

[MBCM19] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A Comparative Study of LPWAN Technologies for Large-Scale IoT Deployment", *ICT Express*, vol. 5, no. 1, pp. 1–7, 2019.

[MBYX22] Y. Mashayekhy, A. Babaei, X.-M. Yuan, and A. Xue, "Impact of Internet of Things (IoT) on Inventory Management: a Literature Survey", *Logistics*, vol. 6, no. 2, p. 33, 2022.

[MK19] A. Madhu and S. Kumaraswamy, "Data Augmentation Using Generative Adversarial Network for Environmental Sound Classification", in *2019 27th European Signal Processing Conference (EUSIPCO)*, IEEE, 2019, pp. 1–5.

[MK20] D. L. Mai and M. K. Kim, "Multi-Hop LoRa Network Protocol With Minimized Latency", *Energies*, vol. 13, no. 6, p. 1368, 2020.

[MLC23] J. Miquel, L. Latorre, and S. Chamaillé-Jammes, "Energy-Efficient Audio Processing at the Edge for Biologging Applications", *Journal of Low Power Electronics and Applications*, vol. 13, no. 2, p. 30, 2023.

[MLR22] M. Mishra, P. B. Lourenço, and G. V. Ramana, "Structural Health Monitoring of Civil Engineering Structures by Using the Internet of Things: a Review", *Journal of Building Engineering*, vol. 48, p. 103 954, 2022.

[MMBB19] P. Mayer, M. Magno, T. Brunner, and L. Benini, "LoRa vs. LoRa: In-Field Evaluation and Comparison For Long-Lifetime Sensor Nodes", in *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI)*, IEEE, 2019, pp. 307–311.

[MMVP15] B. Martinez, M. Monton, I. Vilajosana, and J. D. Prades, "The Power of Models: Modeling Power Consumption for IoT Devices", *IEEE Sensors Journal*, vol. 15, no. 10, pp. 5777–5789, 2015.

[MPB19] S. Martiradonna, G. Piro, and G. Boggia, "On the Evaluation of the NB-IoT Random Access Procedure in Monitoring Infrastructures", *Sensors*, vol. 19, no. 14, p. 3237, Jul. 2019, ISSN: 1424-8220. DOI: 10.3390/s19143237. [Online]. Available: http://dx.doi.org/10.3390/s19143237.

[MT22] A. Mohapatra and T. Trinh, "Early Wildfire Detection Technologies in Practice — a Review", *Sustainability*, vol. 14, no. 19, p. 12 270, 2022.

[Nor16] A. Nordrum, "The Internet of Fewer Things", *IEEE Spectrum*, vol. 53, no. 10, pp. 12–13, 2016.

[PJJ18] S. Popli, R. K. Jha, and S. Jain, "A Survey on Energy Efficient Narrowband Internet of Things (NBIoT): Architecture, Application and Challenges", *IEEE Access*, vol. 7, pp. 16 739–16 776, 2018.

[PMdA+22] L. Prade, J. Moraes, E. de Albuquerque, D. Rosário, and C. B. Both, "Multi-Radio and Multi-Hop LoRa Communication Architecture for Large Scale IoT Deployment", *Computers and Electrical Engineering*, vol. 102, p. 108 242, 2022.

[PMK+17] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey", *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2117–2157, 2017. DOI: 10.1109/COMST.2017.2728092.

[PMMB18] R. Piyare, A. L. Murphy, M. Magno, and L. Benini, "On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT", *Sensors*, vol. 18, no. 11, p. 3718, 2018.

[PMR+15a] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the Coverage of LPWANs: Range Evaluation and Channel Attenuation Model for LoRa Technology", in *ITS Telecommunications (ITST), 2015 14th International Conference on*, IEEE, 2015, pp. 55–59.

[PMR+15b] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of lpwans: Range evaluation and channel attenuation model for lora technology", in *2015 14th international conference on its telecommunications (itst)*, IEEE, 2015, pp. 55–59.

[PSL+19] A. Pastor-Aparicio, J. Segura-Garcia, J. Lopez-Ballester, S. Felici-Castell, M. Garcia-Pineda, and J. J. Perez-Solano, "Psychoacoustic Annoyance Implementation with Wireless Acoustic Sensor Networks for Monitoring in Smart Cities", *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 128–136, 2019.

[RAA22] E. F. I. Raj, M. Appadurai, and K. Athiappan, "Precision farming in modern agriculture", in *Smart Agriculture Automation Using Advanced Technologies: Data Analytics and Machine Learning, Cloud Architecture, Automation and IoT*, Springer, 2022, pp. 61–87.

[RC15] M. A. Razzaque and S. Clarke, "Smart Management of Next Generation Bike Sharing Systems using Internet of Things", in *2015 IEEE First International Smart Cities Conference (ISC2)*, IEEE, 2015, pp. 1–8.

[RDRB18] Y. Roth, J.-B. Doré, L. Ros, and V. Berg, "The Physical Layer of Low Power Wide Area Networks: Strategies, Information Theory's Limit and Existing Solutions", in *Advances in Signal Processing: Reviews, Vol. 1, Book Series*, Aug. 2018. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01872023.

[RJN+17] S. Ramnath, A. Javali, B. Narang, P. Mishra, and S. K. Routray, "IoT Based Localization and Tracking", in *2017 International Conference on IoT and Application (ICIOT)*, IEEE, 2017, pp. 1–4.

[RLHH21] M. Rademacher, H. Linka, T. Horstmann, and M. Henze, "Path Loss in Urban LoRa Networks: a Large-Scale Measurement Study", in *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, IEEE, 2021, pp. 1–6.

[RMG15] R. Ratasuk, N. Mangalvedhe, and A. Ghosh, "Overview of LTE Enhancements for Cellular IoT", in *2015 IEEE 26th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*, IEEE, 2015, pp. 2293–2297.

[RMZ+16] R. Ratasuk, N. Mangalvedhe, Y. Zhang, M. Robert, and J.-P. Koskinen, "Overview of Narrowband IoT in LTE Rel-13", in *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, IEEE, 2016, pp. 1–7.

[RVMG16] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "NB-IoT System for M2M Communication", in *Wireless Communications and Networking Conference (WCNC), 2016 IEEE*, IEEE, 2016, pp. 1–5.

[SCP17] H. Sallouha, A. Chiumento, and S. Pollin, "Localization in Long-Range Ultra Narrow Band IoT Networks Using RSSI", in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6. DOI: 10.1109/ICC.2017.7997195.

[SGA+22] R. Singh, A. Gehlot, S. V. Akram, A. K. Thakur, D. Buddhi, and P. K. Das, "Forest 4.0: Digitalization of Forest Using the Internet of Things (IoT)", *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 8, pp. 5587–5601, 2022.

[SGIB20] H. H. R. Sherazi, L. A. Grieco, M. A. Imran, and G. Boggia, "Energy-Efficient LoRaWAN for Industry 4.0 Applications", *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 891–902, 2020.

[Sha20] A. Shah, "Modern Technology for Sustainable Forest Management in India", *Observer Research Foundation*, vol. 382, pp. 1–14, 2020.

[Skl+01] B. Sklar *et al.*, *Digital Communications*. Prentice Hall Upper Saddle River, NJ, USA, 2001, vol. 2.

[Som11] A. Somov, "Wildfire Safety with Wireless Sensor Networks", *EAI Endorsed Transactions on Ambient Systems*, vol. 1, no. 1, 2011.

[STB+17] B. Sartori, S. Thielemans, M. Bezunartea, A. Braeken, and K. Steenhaut, "Enabling RPL Multihop Communications Based on LoRa", in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, IEEE, 2017, pp. 1–8.

[STD20] L. Sciullo, A. Trotta, and M. Di Felice, "Design and Performance Evaluation of a LoRa-based Mobile Emergency Management System (LOCATE)", *Ad Hoc Networks*, vol. 96, p. 101 993, 2020.

[TCLW19] B. Thoen, G. Callebaut, G. Leenders, and S. Wielandt, "A Deployable LPWAN Platform for Low-Cost and Energy-Constrained IoT Applications", *Sensors*, vol. 19, no. 3, 2019, ISSN: 1424-8220. DOI: 10.3390/s19030585. [Online]. Available: https://www.mdpi.com/1424-8220/19/3/585.

[TCW+17] Y. Tsang, K. Choy, C.-H. Wu, G. Ho, H. Lam, and P. Koo, "An IoT-Based Cargo Monitoring System for Enhancing Operational Effectiveness Under a Cold Chain Environment", *International Journal of Engineering Business Management*, vol. 9, p. 1 847 979 017 749 063, 2017.

[TJY+20] H. P. Tran, W.-S. Jung, T. Yoon, D.-S. Yoo, and H. Oh, "A Two-Hop Real-Time LoRa Protocol for Industrial Monitoring and Control Systems", *IEEE Access*, vol. 8, pp. 126 239–126 252, 2020.

[TJYO22] H. P. Tran, W.-S. Jung, D.-S. Yoo, and H. Oh, "Design and Implementation of a Multi-Hop Real-Time LoRa Protocol for Dynamic LoRa Networks", *Sensors*, vol. 22, no. 9, p. 3518, 2022.

[Van17] L. Vangelista, "Frequency Shift Chirp Modulation: the LoRa Modulation", *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1818–1821, 2017.

[VLC+22] J. Van Mulders, G. Leenders, G. Callebaut, L. De Strycker, and L. Van der Perre, "Aerial Energy Provisioning for Massive Energy-Constrained IoT by UAVs", in *ICC 2022-IEEE International Conference on Communications*, IEEE, 2022, pp. 3574–3579.

[VVV+10] R. J. Vullers, R. Van Schaijk, H. J. Visser, J. Penders, and C. Van Hoof, "Energy Harvesting for Autonomous Wireless Sensor Networks", *IEEE Solid-State Circuits Magazine*, vol. 2, no. 2, pp. 29–38, 2010.

[WCFS20] S.-Y. Wang, J.-E. Chang, H. Fan, and Y.-H. Sun, "Performance Comparisons of NB-IoT, LTE Cat-M1, Sigfox, and LoRa Moving at High Speeds in the Air", in *2020 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, 2020, pp. 1–6.

[WKL+16] A. J. Wixted, P. Kinnaird, H. Larijani, A. Tait, A. Ahmadinia, and N. Strachan, "Evaluation of LoRa and LoRaWAN for Wireless Sensor Networks", in *2016 IEEE SENSORS*, IEEE, 2016, pp. 1–3.

[WLA+17] Y. .-. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, J. Bergman, and H. S. Razaghi, "A Primer on 3GPP Narrowband Internet of Things", *IEEE Communications Magazine*, vol. 55, no. 3, pp. 117–123, 2017. DOI: 10.1109/MCOM.2017.1600510CM.

[ZLB17] M. Zappatore, A. Longo, and M. A. Bochicchio, "Crowd-Sensing our Smart Cities: a Platform for Noise Monitoring and Acoustic Urban Planning", *Journal of Communications Software and Systems*, vol. 13, no. 2, pp. 53–67, 2017.

[ZLS+19] G. Zhu, C.-H. Liao, T. Sakdejayont, I.-W. Lai, Y. Narusue, and H. Morikawa, "Improving the Capacity of a Mesh LoRa Network by Spreading-Factor-Based Network Clustering", *IEEE Access*, vol. 7, pp. 21 584–21 596, 2019.

[ZLSN15] C. Zhu, V. C. Leung, L. Shu, and E. C.-H. Ngai, "Green Internet of Things for Smart World", *IEEE access*, vol. 3, pp. 2151–2162, 2015.

[ZTDW19] W. Zhou, Z. Tong, Z. Y. Dong, and Y. Wang, "LoRa-Hybrid: a LoRaWAN Based Multihop Solution for Regional Microgrid", in *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, IEEE, 2019, pp. 650–654.

[ZZDB15] L. Zhang, J. Zhang, Z.-y. Duan, and D. Bryde, "Sustainable Bike-Sharing Systems: Characteristics and Commonalities Across Cases in Urban China", *Journal of Cleaner Production*, vol. 97, pp. 124–133, 2015.

List of Publications, Presentations and Tutorials

Articles in Internationally Reviewed Academic Journals

G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling for Remote Sensing," MDPI Sensors, vol. 23, no. 11, p. 4994, May 2023, doi: 10.3390/s23114994. [LCO+23]

G. Leenders, G. Callebaut, G. Ottoy, L. Van der Perre and L. De Strycker, "Multi-RAT for IoT: The Potential in Combining LoRaWAN and NB-IoT," IEEE Communication Magazine, vol. 59, no. 12, pp. 98-104, December 2021, doi: 10.1109/MCOM.008.2100382. [LCO+21]

G. Callebaut†, **G. Leenders**†, J. Van Mulders, G. Ottoy, L. De Strycker, and L. Van der Perre, "The Art of Designing Remote IoT Devices — Technologies and Strategies for a Long Battery Life," MDPI Sensors, vol. 21, no. 3, p. 913, January 2021, doi: 10.3390/s21030913. [CLV+21]

B. Thoen†, G. Callebaut†, **G. Leenders** and S. Wielandt, "A Deployable LPWAN Platform for Low-Cost and Energy-Constrained IoT Applications," MDPI Sensors, vol. 19, no. 3: 585, January 2019, doi: 10.3390/s19030585. [TCLW19]

Articles in International Conferences and Symposia

G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "Multi-RAT IoT – What's to Gain? An Energy-Monitoring Platform," IEEE 97th Vehicular Technology Conference (VTC2023-Spring), 2023, doi: 10.1109/VTC2023-Spring57618.2023.10199622. [LCVD23]

G. Leenders, G. Ottoy, G. Callebaut, L. Van der Perre and L. De Strycker, "An Energy-Efficient LoRa Multi-Hop Protocol through Preamble Sampling," IEEE Wireless Communications and Networking Conference, 2023, doi: 10.1109/WCNC55385.2023.10118770. [LOC+23]

J. Cappelle, G. Ottoy, S. Goossens, H. Deprez, J. Van Mulders, **G. Leenders** and G. Callebaut, "IoT with a Soft Touch: A Modular Remote Sensing Platform for STE(A)M Applications," IEEE Sensors Applications Symposium, 2022, doi: 10.1109/SAS54819.2022.9881367. [COG+22]

G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "Tracking Rental Bikes in Smart Cities: a Multi-RAT Approach," Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux, 2022. [LCVD22]

J. Cappelle, J. Van Mulders, S. Goossens, **G. Leenders** and L. Van der Perre, "Strategies for Increasing Longevity of IoT Devices," Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux, 2022. [CVG+22]

G. Leenders, G. Ottoy and G. Callebaut, "Dramco Uno: A Low-Entry IoT Learning Platform for STE(A)M-Oriented Education," Proceedings of the 2022 Symposium on Information Theory and Signal Processing in the Benelux, 2022. [LOC22]

J. Van Mulders, **G. Leenders**, G. Callebaut, L. De Strycker and L. Van der Perre, "Aerial Energy Provisioning for Massive Energy-Constrained IoT by UAVs," IEEE International Conference on Communications (ICC), 2022, doi: 10.1109/ICC45855.2022.9838284. [VLC+22]

G. Leenders, G. Callebaut, L. Van der Perre and L. De Strycker, "An Experimental Evaluation of Energy Trade-Offs in Narrowband IoT," 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), New Orleans, LA, USA, 2020, pp. 1-5, doi: 10.1109/WF-IoT48130.2020.9221010. [LCVD20]

G. Leenders, C. Buyle, G. Callebaut, and L. Van Der Perre, "An Experimental Evaluation of Power Optimization in NB-IoT" Proceedings of the 2019 Symposium on Information Theory and Signal Processing in the Benelux, 2019. [LBCV19]

G. Callebaut, **G. Leenders**, C. Buyle, S. Crul and L. Van der Perre, "LoRa Physical Layer Evaluation for Point-to-Point Links and Coverage Measurements in Diverse Environments," European Conference on Networks and Communications, 2019. [CLB+19]

S. Crul, **G. Leenders** and L. Van der Perre, "Glue-and-Play Sensing Solution for Remotely Monitoring Drinking Frequency of Horses," IEEE Sensors Applications Symposium, 2019, doi: 10.1109/SENSORS43011.2019.8956605. [CLV19]

G. Leenders, G. Callebaut, G. Ottoy and L. Van der Perre, "Versatile module for connecting sensors via Low Power Wide Area Network (LPWAN)," IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2017. [LCOV17]

Presentations and Tutorials Presented in International Conferences and Symposia

L. Van der Perre, S. Goossens, H. Deprez, J. Van Mulders, S. Crul, B. Thoen, M. Alleman, **G. Leenders**, G. Callebaut, G. Ottoy, "IoT with a Soft Touch: Connecting for a Sustainable Society", presented at Semicon Europe, November 2019, Munchen Germany.

G. Leenders, G. Callebaut, "Versatile module for connecting sensors via Low Power Wide Area Network (LPWAN)", presented at Imec Wireless Community, January 2018, Leuven Belgium.

G. Callebaut, **G. Leenders**, "Low-power wireless technologies for connecting embedded sensors in the IoT: A journey from fundaments to hands-on", presented at IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, October 2017, Montreal Canada.

FACULTY OF ENGINEERING TECHNOLOGY
DEPARTMENT OF ELECTRICAL ENGINEERING
DRAMCO-WAVECORE
Gebroeders De Smetstraat 1
B-9000 Ghent