UAB

Universitat Autònoma de Barcelona

DEPARTMENT OF MICROELECTRONICS AND ELECTRONIC SYSTEMS

Exploiting Spatio-Temporal Correlations
for Energy Management Policies

Ph.D. Candidate:

BORJA MARTÍNEZ HUERTA

Supervisor:

Dr. Ignasi Vilajosana

Departmental Supervisor:

Dr. Màrius Montón

DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE

MMXV

Dedicated to my parents,
for their unconditional support.

Acknowledgments

I consider Dr. Michele Rossi to be one of my mentors. I am grateful for his useful advice, and for allowing me work with him at the University of Padova. I am indebted to Dr. Kris Pister, for the opportunity he provided me to work with him at Berkeley. I would also like to thank Dr. Mischa Dohler of King's College, London. This thesis wouldn't be possible without them.

. . .

I would like to thank all the people who have helped me with this dissertation. Special thanks to Sneha for her lovely contribution and to Gaëlle for her patience.

. . .

I would like to thank the people who have inspired me on this long journey. Dr. Màrius Montón, for the many things that I learnt from him, for the constructive discussions, for his support in hard times and, above all, for his friendship. Dr. Xavi Vilajosana, for his guidance, his tireless help and his infinite energy. I would like to express my sincere gratitude to my supervisor and friend, Dr. Ignasi Vilajosana.

. . .

My last words of acknowledgment are to my family. I am privileged and grateful to have two sisters, Monica and Leticia, who never stopped encouraging me. For your complete trust, thank you. And finally my parents, Pedro and Mª Teresa. Their sacrifices and love got me where I am today. I owe them more than I will ever be able to put down in words. To them, I dedicate this work.

Abstract

We are living in a new era, which is characterized by the omnipresence of smart, networked devices. The developing Internet of Things is profoundly transforming both global industry and human lives. Hardware integration, along with the ability to seamlessly communicate over the internet, has allowed millions of embedded objects to connect and interact on an unprecedented scale. The ubiquitous presence of embedded computing devices, combined with their sensing and communicating capabilities, is increasing the amounts of data captured on a massive scale.

As a result of the expanding IoT, the number of connected devices is increasing exponentially and will soon generate a problem of scalability, related mostly to their energy dependence. Many devices will be embedded in the environment, in places that are inaccessible or expensive to connect with wires, making them resource-constrained. Most importantly, battery replacements for thousands of devices are inconceivable. Maintenance and intervention costs can limit the advance of this new paradigm. Therefore, one of the challenges in ensuring the massive expansion of wireless sensing devices is reducing their cost in terms of energy. Clearly, novel methods are required for addressing this change.

Spatio-temporal correlations are essential in many different fields. Thus, it is quite reasonable to assume that contextual information can be exploited within this emerging paradigm. Under this hypothesis, the present study provides a systematic approach to defining *Energy Efficiency Policies* for wireless sensor devices, based on the analysis of *Spatio-Temporal Correlations*.

To this end, the present work is structured in two parts. First, we address the necessity of an accurate energy profiling model for wireless sensing devices. We have formalized a generic consumption model to profile the energy utilization of low-power embedded devices. The obtained results stress the importance of understanding the cycles of operation involved in embedded tasks. The second part of this dissertation demonstrates the applicability of spatio-temporal correlation analysis as a tool for defining energy efficiency policies. This hypothesis has been investigated from three different perspectives: a) energy harvesting, b) data compression and c) contextual data analysis. The correct analysis and policy definition from these three perspectives provides important energy and cost reduction opportunities.

In conclusion, all the studied methods proved to be effective for defining and validating energy policies. The proposed strategies help designers to parameterize and customize platforms for their application during the design phases, and hence the time-to-market of new products is reduced while an optimal tradeoff is ensured among cost, functionality and life expectancy.

Contents

List of Figures

List of Tables

Prologue

THE library of Alexandria, abuzz with conversations, lectures and other activities on most days, was quiet. It was a hot day; one of the hottest that Alexandria had seen in many years. Scholars, scientists, philosophers, mathematicians, artists, historians, students, teachers and scribes, had arrived at the library that morning and then within minutes, decided to shirk work for the day and go to the seaside for a swim. Only one of the scholars, the head librarian, Eratosthenes, had been in disagreement with this plan.

The last month had been busy – summer solstice had come and gone and a trade ship from Cyrene had arrived carrying books by Homer. As was the command of the king and the rule of the land, each of the books had been confiscated and then painstakingly copied at the library, so that the originals could have a new home. The copies were so perfect that the captain of the ship hadn't realized that he was being returned something he had never owned.

Eratosthenes, who had a particular fondness for the Odyssey, had orchestrated the creation of a new section in the library for Homer's works. He sat in this room now, by a window overlooking the port, glad that he had the space to himself.

In a rare moment of indulgence – it wasn't often that he allowed himself to reread his own work – he was busy copying something he had written a few years ago; Chronographies. This was a text so accurate in its history of key events that Ptolemy III Euergetes, the king of Alexandria and father of Eratosthenes' primary pupil, Philopator, had read it a few years ago and urged him to accept the prestigious role of head librarian at the library of Alexandria.

Eratosthenes was writing the name of the 37th king of the Egyptian Thebes, when a voice jolted him out of his concentration. It was a voice full of energy and curiosity, the voice of a young prince still not sullied by the weight of the responsibility that would come to him.

'Master!' the voice said, 'I have come back from Syene.'

'Oh, it's you, Philopator. Why aren't you at the sea with the others?'

'I had to come and talk to you Master. No one else will believe me, not even my mother!'

'What are you talking about child? Calm your mind and tell me.'

'Well, last week, one afternoon, we were on the island in Syene. Mother was thirsty so she asked me to get her some water from the well. But Master, when I looked inside the well, there was no shadow. When I told mother and the others, they said I must be seeing things or that I must have done something to displease the Gods.'

'Philopator, how is that possible? You know that everything casts a shadow. We studied it together, don't you remember?'

'I remember, Master, but I saw what I saw. And that day at noon, it was the 21st of June –mother told me that is the day of the solstice– there was no shadow in the well,' Philopator paused, 'Master, you believe me, don't you?'

Eratosthenes looked at him. He knew that the prince, although wild in his imaginations, would not make up a tale or utter a lie to a teacher he so respected.

'Alright, Philopator, I will believe you. I do not know why you saw what you saw, but I give you my word that I will try to find a reason for this strange occurrence. Now leave me with my work and go and play with the others. May the Gods be with you.'

Satisfied that, at last, he had found a patient listener, Philopator ran out, leaving the head librarian in solitude once again.

For the rest of the day, Eratosthenes found that he couldn't concentrate on editing the Odyssey anymore. His student, at the age of ten, had posed him a question that he couldn't answer and it consumed him.

· · ·

A year passed and Eratosthenes found that he couldn't focus on anything for too long. Why did the sunlight bound off the water in Syene and not in Alexandria? Why was there always a shadow in Alexandria? Why wasn't anyone except Philopator shocked by this?

On one such day of distractions, the day of the solstice, Eratosthenes found that he was in the company of young Philopator again. They had come out of the library for a walk because neither of them had been able to concentrate on their lesson, the teacher, annoyed that he still hadn't found an answer to a child's question and the student, too hot to study.

After they walked for a while, they arrived at a well. Philopator stopped.

'Master, I'm thirsty, shall we get some water?'

He had already picked up a stick so he could draw out the water from the well using the pail that lay by its side. It was midday, the stick had a shadow.

'Philopator!' Eratosthenes exclaimed, 'Stay there, stay exactly as you are, I'll be right back.'

He came back after a few minutes, bringing with him his measuring instruments.

'Look, Philopator, the sun is above us but the stick casts a shadow at an angle. Right now, the sun is above Syene, on the island you were last year with your mother, but I am positive that the same stick would cast no shadow there. Come, let us make haste and return to the library. I think I know now what I must do.'

Philopator's face had lit up, as though a memory of the past had resurfaced in his mind. 'I knew you would find an answer, Master. But what is it that you have to do?'

Eratosthenes smiled, 'My dear Philopator, I believe I am going to find out the size of our beloved Earth.'

…

Centuries later, an explorer called Christopher Columbus would refer to the works of Eratosthenes and choose to ignore the scholar's calculations of the size of the Earth. If he hadn't done so, he might have arrived at the destination he had originally intended to arrive at – India.

"Journeys Around The Earth" — Sneha Nagesh

Introduction

Historically, spatio-temporal correlations have been very successful in providing insights into nature. Egyptian mythology, inspired by the cycles of nature, saw time in the present as a series of recurring patterns, whereas the earliest periods of time were linear. Undoubtedly, the yearly cycle of the Nile was essential to the foundations of Egyptian Civilization. Notably, it seems to have also been present in the first steps of science when, in addition to observations of the moon and stars, the periodic flood was used to delimit the beginning of the year, being essential in the elaboration of the first calendars.

In the 20th century, spatial-temporal correlations have been successfully used in many fields of physics, from fluid characterization [40] to a better understanding of our universe's expansion [54]. Correlations are essential in many branches of science, from medicine to the economy, and their importance was synthesized by W.R. Tobler in his expression of the so-called first law of geography: "everything is related to everything else, but near things are more related than distant things"[70].

With the arrival of the 21st century, we have seen the rise of more and more integrated technologies, new wireless communication capabilities and big data analytics: we are facing a new era enabled by the mobile internet. The vast deployments of the so-called Internet of Things (IoT) [26] are profoundly transforming global industry. Hardware miniaturization, communications standardization and analytics are allowing billions of objects to connect and interact [7].

IoT devices, growing analytics and big data capabilities have the potential for enhancing industry and service operations in multiple sectors. Unquestionably, the ubiquitous presence of IoT devices, combined with their sensing and communicating capabilities, is expanding the opportunity to capture data on an unprecedented scale [31]. Spatio-temporal correlations can now be exploited on a wider scale and are destined to be key in lowering maintenance and operational costs for many industries. Moreover, at the same time, spatio-temporal correlations can be used to seamlessly integrate IoT devices into our environment and make them more efficient and affordable.

ONE of the challenges in ensuring the massive expansion of wireless sensing devices is to reduce their power consumption. If the number of devices keeps increasing as predicted [Dodson2003], the IoT era will soon suffer from a problem of scalability. Maintenance, interventions and battery replacements, among others, will become prohibitively expensive as the number of devices increases. For this reason, new design methodologies should lean toward more *Sustainable IoT Devices.*

Most of the time, the energy efficiency of IoT devices is driven by new microcontroller designs and new radio technologies. In recent years, several strategies have been developed to prolong the lifetime of sensor nodes that exploit the data they capture. These include processing techniques such as data aggregation [33], distributed or temporal compression [68], as well as battery replenishment through energy harvesting [79]; but little has been done to help us better understand how to use contextual information to improve their performance.

In the present work, we focus on the contextual information of sensors, i.e., the position of the devices, environmental conditions, their relation to other individual sensors in the network and the characteristics of the physical magnitudes sensed. Our aim is to use this information to develop energy efficiency strategies. Therefore, the general objective of this dissertation is to demonstrate that contextual information can be a very valuable tool for reducing the power consumption of a specific IoT device in a network.

More formally, we aim to demonstrate the value of exploiting spatio-temporal correlations to improve the energy efficiency of a system by adapting its behavior to its specific environment. This dissertation will show how to accomplish such an improvement in several ways, as is illustrated in the following sections. Special attention is paid to the tradeoff and constraints imposed by contextual information, which will be highlighted and discussed throughout the present work. However, specific technological limitations or features are intentionally kept out of the present dissertation.

More specifically, the opportunity to increase efficiency relies on being able to identify and characterize the spatial correlations in the sensor's environment, particularly in terms of how they relate to the application of interest. When the observed patterns are combined with the temporal correlations in the sensor's activity, they can be used for novel strategies to create efficient energy management policies. The latter applies to multiple areas of IoT development. In this context, to clarify how the present work contributes to the overall progress of the state of the art, three main topics are addressed:

- **Self-powered sensors:** Spatio-temporal correlations occur naturally in the behavior of the energy source. They prove useful in characterizing the energy scavenging opportunities and coupling them with energy demand.

- **Data Compression:** Compression has been used traditionally for reducing the amount of data communicated through the wireless channel, and thus the energy required. The temporal correlations observed in the acquired data have proven to be essential in achieving efficient compression.

- **Lean sensing**: Spatio-temporal correlations uncover underlying information in the acquired data, which can be smartly exploited to optimize energy consumption while minimizing the impact on accuracy. They prove essential for the exploration of new strategies that reduce the amount of data acquired while, at the same time, they maintain an accurate representation of the context.

Structure and Main Contributions

This thesis is organized in two parts. The first part of the thesis (Chapter 2) is dedicated to the energy profiling of industrial wireless sensing devices. The second part (Chapters 3, 4 and 5) introduces new developments and presents a discussion of how to apply the proposed methodology in three different areas of application.

Going more into detail about Chapter 2, a methodology based on a parameterizable model is presented. The methodology aims to reduce technology adoption and integration risks, as energy consumption can be estimated precisely without the need for prototyping or building actual devices. Furthermore, the model allows for a better understanding of the energy usage. The proposed methodology is based on the identification and parameterization of the main sources of energy consumption. Once they are identified, optimal system design and implementation ensures minimized costs and risk. Moreover a systematic analysis like the one proposed here enables engineers to make better-informed decisions in the early stages of development.

More specifically, Chapter 2 contributes to the overall progress of the state of the art by:

- Defining a new systematic methodology at the pre-deployment stages, one that is especially suited to industrial applications. This methodology is intended to allow for a priori definition of the system architecture and to support design decision

- Introducing a system-level perspective to adjust the energy consumption of applications running on wireless devices, balancing the cost of communication, processing and data acquisition subsystems.

- The methodology is empirically validated against different industrial applications. The presented experimental results demonstrate the model's suitability for different scenarios and platforms.

The main contents of this chapter have been compiled for publication under the title *"The Power of Models: Modeling Power Consumption for IoT devices."* [REF]. As far as we know, this is the first article presenting a formalized model that takes into account the three cornerstones of any industrial wireless sensing applications: standard networking technologies, data acquisition and local processing.

. . .

The second part (Chapters 3, 4 and 5) introduces new developments and presents a discussion of how to apply the proposed methodology in three different areas of application, where the energetic model presented in Chapter 2 is used as a basis to determine the benefits, drawbacks and trade-off of several proposed energy efficiency strategies. Spatio-temporal correlations are used to adjust the model according to energy availability (Chapter 3), data characteristics (Chapter 4) and service provided (Chapter 5).

The focus of Chapter 3 is twofold. Firstly, the study utilizes the model presented in Chapter 2 and applies it to two real-world industrial applications: the predictive analysis of rotating machines and structural health monitoring. The study illustrates the value of the model as a tool for supporting application design. More specifically, the presented model is used to understand the trade off and limitations of the sensing devices in relation to the available energy and the desired performance of the monitoring system. Secondly, as result of the acquired understanding, an appropriate strategy is proposed for minimizing the energy consumption of the nodes by processing all the data locally. The study shows the effectiveness of the proposed strategies in their specific scenarios.

Chapter 3 contributes to the overall progress of the state of the art in the following way.

- It presents a new modeling methodology that allows us to dimension the device's energy needs according to its requirements for communication, data acquisition and processing. The aim of this methodology is to gain an a priori understanding of the tolerance margins for a device that is self-powered by means of a harvester. This approach ensures the right dimensioning of the system and enables potential cost optimization for product development. Additionally, the impact of other modes of operation can be evaluated at the prototyping stages.

- By using two real-world case studies as a guide, this chapter illustrates the potential application adjustments that balance communication costs, processing and data acquisition subsystems, such that they are all in accordance with the amount of harvested energy.

- This chapter demonstrates the energy self-sustainability of sensing devices by using low-power wireless protocols in combination with energy scavengers. We show the utility of performing an accurate sensitivity analysis during product development.

The main ideas discussed throughout this chapter have been published under the title *"When Scavenger Meets Industrial Wireless"* [REF]. This work was later expanded into a more detailed article under the title *"Early Scavenger Dimensioning in Wireless Industrial Monitoring Applications"* [REF].

$$\dots$$

In Chapter 4, we concentrate on a strategy that reduces data information at the expense of accuracy. In this way, radio activity is reduced so that power consumption is eventually reduced as well. Specifically, lossy compression mechanisms are proposed as a method for diminishing the amount of data to be communicated. In this case, the model presented in Chapter 2 is utilized to evaluate a number of selected lossy compression methods from an energetic point of view. An extensive analysis is conducted on their performance in terms of compression efficiency, computational complexity and energy consumption. Specifically, Chapter 4 contributes to the overall progress of the state of the art in the following way.

- It provides an in depth performance evaluation of selected lossy compression algorithms for time series. These include linear and autoregressive models, as well as Fourier and Wavelet transforms. This chapter performs a quantitative assessment and trade-off analysis of compression as a tool for energy consumption reduction. We also evaluate compression algorithms for a given reconstruction fidelity, the signal statistics and the hardware characteristics.

- It provides a formal mathematical derivation of underlying processes. They are obtained through numerical fittings and validated against real datasets in order to gauge computational complexity, overall energy consumption and signal representation accuracy of the best performing compression algorithms as a function of the most relevant system parameters.

- Notably, this chapter reveals that compression schemes can be inefficient in terms of energy consumption when accounting for the energy required to compute the compression algorithms. Consequently, the efficiency of compression should always be pre-evaluated.

This chapter has been published integrally under the title *"On the Performance of Lossy Compression Schemes for Energy Constrained Sensor Networking"* [REF]

$$\dots$$

In Chapter 5, we propose going one step further by introducing contextual information (application dependent) in power consumption optimization, especially for applications where local

processing (Chapter 3) and data compression (Chapter 4) are not suitable. To this end, the model presented in Chapter 2 is utilized to evaluate the impact of a number of optimization strategies designed at the application level. They include adaptive sampling and, more specifically, using spatio-temporal correlations for adapting sensor behavior in order to balance and reduce the power consumption of a sensor network. Specifically, Chapter 5 contributes to the overall progress of the state of the art in the following way.

- This chapter investigates the impact of new data capturing strategies that are adapted to contextual information. For instance, adaptive or dynamic sampling rates are proposed for developing lean-sensing applications. The work departs from a system-level application analysis in order to define energy efficient strategies that do not compromise application performance.

- This chapter establishes a basis for introducing contextual information into energy management strategies through a real example. Specifically, the observed spatio-temporal correlations at the system level allow us to define specific metrics for system optimization. The observed benefits are based on perceptible user experiences and other quality parameters, but they quantitatively reduce the power consumption of the overall system. As a guiding real-world example, an extensive, large-scale analysis is made on a parking monitoring application in two different cities.

- Remarkably, this chapter reveals how contextual information can be optimally applied for reducing the energy consumption of IoT devices without compromising overall system behavior, especially in terms of how it affects the user's experience.

The topics presented in this chapter constitute the basis of an article entitled *"Exploiting contextual information in low-power sensing"*[REF]. An additional discussion about the cost-benefit relationship in smart city ecosystems has been published in the article *"Bootstrapping Smart-Cities through a Self-Sustainable Model Based on Big Data Flows"*[REF].

. . .

Finally, the last part summarizes and discusses the main conclusions of the present work.

Chapter 2

Energy Models for WSN Devices

Creating more energy efficient technologies is still far from those ubiquitous deployments that were envisaged (the so-called Internet of Things or, more recently, the Industrial Internet), which will enable optimal industrial operation or contribute to improving social welfare. Viable systems, especially for the industry, are those whose operational costs enable fast returns on investment, even when deployed at a large scale. Battery life remains a major obstacle to viability.

Indeed, today it is possible to build a device which features this industrial wireless performance. However, energy-dimensioning the device in order for it to meet the application requirements is not an easy task. In most cases, designs are guided by worst cases scenarios for hardware energy consumption, without even considering the duty-cycled behavior of the different subtasks in the target application. For efficient energy management, it is essential to understand dependence on the main application parameters and their interrelations, two factors which severely compromise the reliability required for industrial applications. This is even more critical in the case of wireless sensors, as the energy is a scarce resource.

As a natural consequence, this work assumes that to develop extremely low power devices, a system-wide characterization is fundamental for achieving the specifications at a reasonable cost. Thus, this chapter proposes a methodology to aid engineers in understanding the energy life-cycle within the application, which will enable them to determine tolerance margins and trade-offs.

2.1   Introduction: Energy Flow in Low-Power Embedded Systems

L OW-POWER Wireless Sensor Networks are made up of tiny sensor devices with communication capabilities. Sensors may also have a kind of rechargeable battery onboard (hereafter referred to as an energy buffer) and dedicated circuitry, whose function is to scavenge energy from the surrounding physical environment. Fig. 2.1 shows a schematic diagram of such a configuration.
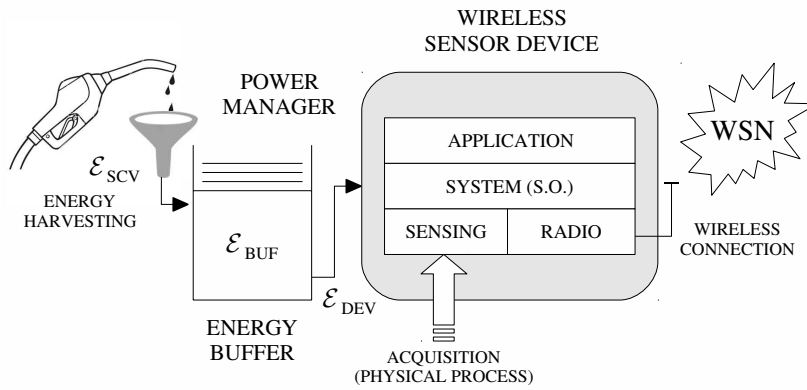
Figure 2.1. *Generic wireless sensor device model.*

This figure shows the three main blocks to consider when addressing the problem of energy consumption for these specific devices:

- **Power manager:** captures ambient energy surrounding the device and convert it into usable electrical energy. In addition, the energy gathered must be properly adapted to feed the following blocks [37].

- **Energy buffer:** a battery or any other device capable of storing and releasing energy [13].

- **Embedded device:** the end-user. Responsible for all system-application functions and management, which require energy for their operation [10].

In this framework, Eq. (2.1) establishes the *energy flow model* [37] of a networked embedded device:

$$\mathcal{E}_{BUF}^{(t=0)} + \int_{\tau=0}^{t} \mathcal{P}_{SCV}(\tau)d\tau \geq \int_{\tau=0}^{t} \mathcal{P}_{DEV}(\tau)d\tau = \mathcal{E}_{DEV}(t) \tag{2.1}$$

In this general model, the energy initially stored in the buffer $\mathcal{E}_{BUF}$ and the additional energy obtained from the medium $\mathcal{E}_{SCV}$ must be greater than the energy that the device requires to operate $\mathcal{E}_{DEV}$ throughout its whole life. A natural constraint follows from *energy causality*, which dictates that energy cannot be used before it is available [34]. Therefore, the condition in Eq. (2.1) must hold strictly $\forall t$.

Recently, the topic of *neutral design policies* is thriving in the WSN research community (see [15] and citations included). The concept of neutrality accounts for the fact that the energy used over the long term should be, at most, equal to that harvested ($\mathcal{E}_{SCV}(t) \geq \mathcal{E}_{DEV}(t), t \to \infty$). In other words, the energy initially stored $\mathcal{E}_{BUF}^{(t=0)}$ in the first part of Eq. (2.1) is negligible after some time running. This is a general condition for the sensors to be energetically self-sufficient; that is, the devices may ideally last unattended for an unlimited period of time [36].

In computing, the producer-consumer problem (also known as the bounded-buffer problem) is a classic problem of multi-process synchronization. The problem involves two processes, the producer and the consumer, who share a common, fixed-size buffer used as a queue. The producer's job is to generate a piece of data, put it into the buffer and start again. At the same time, the consumer is consuming the data, i.e., removing it from the buffer. The goal is to make sure that the producer won't try to add data into the buffer if it's full and that the consumer won't try to remove data from an empty buffer.

Energy flow in a wireless sensor device follows this model closely. However, the consumer's goal is not always to dispatch tasks from the queue as fast as possible. Indeed, sometimes the device will operate with an aggressive configuration, thus maximizing the processing performance and avoiding buffer overflows [72]. Yet, most of the time, the optimization objective will be more conservative in order to optimize power consumption. These alternative cost functions determine the *energy policies* of the device.

Energy policies demand a perfect understanding of the energy flow, that is, how energy is generated, stored and consumed. Thus, one of the main objectives of this work is to convert Eq. (2.1) into a practical model, that is, a model that can be easily simulated and numerically evaluated. This model should provide concrete answers to abstract problem formulations such as energy causality, neutrality or sustainability. This chapter focuses on energy drain, while production and storage will be addressed in Chapter 3. Therefore, in what follows, a system-level model of the device's power consumption is developed.

2.2　Related Work

The use of system-level consumption models to support the design of energy-constrained devices is a rather scantily studied topic. In the past few years, a vast literature has emerged in the field of Wireless Sensor Networks, particularly in regard to the energy characterization of their tiny devices. Most of the work focuses on network activity, i.e., how communication issues affect to the device's power consumption, keeping in mind that each sensor within the network is a member of a complex and interrelated system. Other studies are concerned with the role of processors in energy expenditure, or even with specific details about the sensing process itself. But, within this topic, system level constraints have rarely been addressed. For this reason, the applicability of these approaches is limited in practical designs.

This section reviews in brief some of the most representative works on these subjects. The aim is merely to provide a rough outline of how different perspectives approach modeling. However, the works referred here are important for three main reasons: i) these strategies provide useful insights into some of the design trade-offs; ii) Some of them are used as constituent elements for creating the new methodology; iii) selected works provide further reading about specific questions through included references.

Communication Models

Chen *et al.* [19] collect and review several studies related with network energy optimization. This work classifies technologies into time, frequency, and spatial domains. It briefly describes the main solutions in each domain and provides a better understanding of energy consumption in wireless communications As mentioned above, design decisions cannot be based only on network considerations when dealing with low-power embedded devices. For these reason, the practical value of these approaches is somewhat limited.

Due to its influence on the present work, it is worth noting the proposal by Wang *et al.* [81]. This paper develops a method for estimating the energy spent during the communication process. Neither sampling techniques nor the cost of the application itself are considered. However, it presents a model for the power consumption of networked sensors based on the characterization of the *atomic blocks*, i.e., individual components that are involved in communication. The methodology proposed for building their models has been used as a starting point for the development of the present work.

Processing Models

Regarding processing power, several measurement-based methods can be found in the literature. Most of these models, like those presented in Nikolaidis *et al.* [52], Bazzaz *et al.* [9] and Konstantakos

et al. [41], use data obtained from a physical target device and associate the instructions with the corresponding energy cost. The total energy consumption of the application is the aggregate cost of all executed instructions, which can be calculated by running the application in an emulator. These works focus on the accurate energy profiling of the CPU and processor peripherals (FLASH, RAM, ADCs). Yet none of them consider external components that play a fundamental role in today's embedded devices, mainly communication components.

The main advantage of measurement-based methods is the highly accurate energy estimate that they obtain, which is a result of using actual values measured in the target platform. The model fitting methodology developed in Sec.2.4 takes advantage of this fact and can be considered an inheritor of these methods.

Sensing Models

Some recent approaches to modeling scavenging techniques in industrial wireless applications deal with the sampling energy, although this issue is still far from being addressed in depth. For instance, Torah *et al.* [71] assume a dependence between the harvesting pattern and the application's needs, and they draw-up a best-effort policy: an application wakes up the microcontroller and transmits a packet message when enough energy has been harvested. The sampling contribution is estimated in order to compute the total energy, but this work provides neither a clear modeling of the application's energy requirements nor a detailed network energy consumption analysis.

More related with this work, Konstantakos *et al.* [41] evaluate the cost of capturing the sample from the processor side. The model is based on the type of executed assembly instructions, as well as the number of accesses to the memory and the analog-to-digital converter. This way of evaluating the cost associated with the CPU intervention serves as a model for some parts of the present methodology (Sections 2.3.2 and 2.3.3). However, the power consumption associated with external sensors is not computed.

In general, the energy consumption by the sensor has been underestimated in the literature. Yet, this part can contribute substantially to the device's consumption, as will be demonstrated. This is particularly true for active sensors (those requiring some external excitation).

System Level Considerations

From a different perspective, Lu and Gungor [44] analyze several *system-level design* aspects of wireless embedded systems. This survey identifies the synergies between wireless sensor networks, nonintrusive monitoring based on electrical signals, and fault diagnosis for industrial systems. The main scope is to provide a system overview of applications in WSN architecture. This paper also provides detailed analyses for addressing the real-world challenges in designing and deploying practical

WSNs. These analyses include wireless-link-quality dynamics, noise and interference, communication range and reliability. However, the impact of these system-level issues on energy consumption is not clearly addressed.

$$\cdots$$

As may be inferred from this brief analysis, developed models usually cover only partial areas of the design space. But the energy available is a shared resource of the system, so each player must subsist with their own budget. In general, a systematic study of how energy is distributed in the whole system has been rarely addressed in the literature. The development of the methodology proposed in this chapter was in part motivated by the lack of a rigorous and systematic approach, i.e., one that incorporates a system-wide view toward modeling the energy use of smart-sensor devices. The methodology developed in this chapter is built upon several partial solutions gathered from the literature, while at the same time providing the system-level model with a wider perspective.

2.3   Subsystem Bottom-Up Modelling

The power required to operate a wireless sensor device can be broken down into three main blocks: for data sensing or *acquisition* $\mathcal{P}_{ACQ}$, for data handling or *processing* $\mathcal{P}_{PRC}$, and data communication or *networking* $\mathcal{P}_{NET}$. Additionally, a small fraction of the available resources is intended for system management, such as running a real-time operating system (RTOS) or periodic system wake-ups. The set of all management tasks are collected in this $\mathcal{P}_{SYS}$ contribution. These elements together make up the general expression of a device's power $\mathcal{P}_{DEV}$, as summarized in Eq. (2.2).

$$\mathcal{P}_{DEV} = \mathcal{P}_{NET} + \mathcal{P}_{ACQ} + \mathcal{P}_{PRC} + \mathcal{P}_{SYS} \tag{2.2}$$

Based on the recurrent behavior, most industrial applications can be easily characterized because they follow a common operational pattern: data is acquired by some sensor of the system, processed in a controller unit and, finally, some information is sent through a wireless channel. This process repeats over time, and the role of the duty-cycle is fundamental in the power consumption: the larger the duty cycle, the lower average power.

Fig. 2.2 sketches out the temporal sequence of a typical monitoring application. The system wakes-up periodically with elapsed time of T_{RCD}. For each cycle, three steps are executed: i) capture a set of N_S samples separated by some time interval T_S; ii) process or analyze the *record* [1] of acquired samples; and iii) report gathered data, update server information or trigger an alarm when an anomaly is detected. This requires the periodic transmission of radio messages with some characteristic interval T_{MSG}.

In Fig. 2.2, the vertical dimension represents the instantaneous power consumption of the device. Therefore, shaded areas depict the accumulated energy for each task. $\mathcal{E}_{NET}$ stands for the energy drained for communication tasks, $\mathcal{E}_{ACQ}$ for acquisition and $\mathcal{E}_{PRC}$ for processing. In the background, the operating system or scheduler execute different synchronization and coordination tasks, which may include network management. This systematic activity is carried out within T_{SYS} cycles with an associated energy $\mathcal{E}_{SYS}$. Finally, the dashed line indicates the average power of the device $\bar{P}_{DEV}$.

The proposed model is based on an atomic characterization of each building block, where the instantaneous power consumption is integrated over the duration of the corresponding task; the resulting energy is then averaged out over its characteristic *temporal scale* or period of repetition. The next subsections will go into detail on the analysis of each building block.

---

[1]A record is defined as the process of waking up, taking a set of samples and leave it in memory ready to be processed.
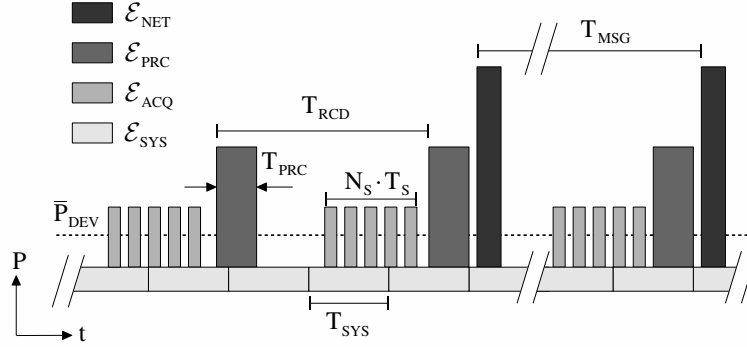
Figure 2.2. *Characteristic time evolution of energy usage split into different components. The vertical dimension represents the instantaneous power consumption. Shaded areas depict the accumulated energy for each task. The dashed line in the figure represents the average power of the device $\bar{P}_{DEV}$.*

2.3.1 Network Models

Point-to-Point Communications

The simplest model for wireless communications consist of an interference-free, single-hop scenario. The Medium Access Control (MAC) layer is idealized, i.e., besides transmission and reception, it does not introduce further energetic inefficiencies due to collisions and idle times for floor acquisition. In this case, the power consumption can be estimated for each device individually. As any attempt at transmission is supposed to arrive to the destination, the model does not need to cope with interferences caused by other devices, congestion or any other collective issues.

Under these assumptions, the average power of the communications block can be expressed by Eq. (2.3) in terms of the energy required to send a radio message $\mathcal{E}_{MSG}$ and the time between consecutive messages $T_{MSG}^{(i)}$. The index i of the summation runs for all messages N_{MSG} on the averaging period.

$$\bar{\mathcal{P}}_{NET} = \sum_{i=0}^{N_{MSG}} \frac{\mathcal{E}_{MSG}}{T_{MSG}^{(i)}} \tag{2.3}$$

The energy per message $\mathcal{E}_{MSG}$ is a parameter that depends only on the specific radio technology. Two key factors play a fundamental role in this contribution: radio power and transmission time. Radio power tends to be maximized to increase the communications range, although it is legally limited in each ISM band. In contrast, transmission time is a parameter determined mainly by the modulation: depending on how a message is spread over time, it results in a complex trade-off between bit-rate (and thus consumption), range, reliability and immunity to interferences . The study of the impact of modulation on radio performance is out of the scope of this work. Yet, regarding the

estimation of power consumption, considering the footprint of modulation on the transmission time can provide a new degree of freedom in the system design flow.

Regarding the time between messages, T_{MSG}^0 can be considered a constant parameter for periodic reporting applications. In this case, Eq. (2.3) reduces to a simpler expression given by Eq. (2.4)

$$\bar{\mathcal{P}}_{NET} = \frac{\mathcal{E}_{MSG}}{T_{MSG}^0} \tag{2.4}$$

In a more general case, sensors generate endogenous traffic, each one according to some distribution or stochastic process. The production rate, characterized by a certain probability distribution (PDF), depends basically on the underlying physical process. In this situation, the time elapsed between consecutive messages in Eq. (2.3) should be characterized by an appropriate statistical estimator (typically the expected value $\widehat{E}[\,]$ of the distribution, defined as $\widehat{T}_{MSG}$), leading to Eq. (2.5). This approximation should be good enough for long-term averaging.

$$\bar{\mathcal{P}}_{NET} \approx \frac{\mathcal{E}_{MSG}}{\widehat{E}\left[T_{MSG}\right]} \doteq \frac{\mathcal{E}_{MSG}}{\widehat{T}_{MSG}} \tag{2.5}$$

The energy cost associated with each transmission may in turn depend on multiple factors:

- **Retransmissions:** some opportunistic approaches just retransmit the same message several times in order to enhance the probability of delivery success. In this case, the energy cost per message is simply multiplied by the number of attempts N_R in Eq. (2.6).

$$\bar{\mathcal{P}}_{NET} = N_R \cdot \mathcal{E}_{MSG}/T_{MSG} \tag{2.6}$$

- **Radio power:** most radio chips allow some control over transmission power, providing an additional trade-off between energy cost and distance range [2]. Typically, output level is selected from among a set of discrete values N_P, leading to a quantized energy scale in Eq. (2.7).

$$\bar{\mathcal{P}}_{NET} = \mathcal{E}_{MSG}^{(N_P)}/T_{MSG} \tag{2.7}$$

- **Spreading factor:** some radio technologies can operate with different *spreading factors* N_{SF}. The spreading factor increases the communication range but lowers the bit-rate of the transmission. As the transmission time increases, more energy is required. This behavior can be easily modeled by some suitable function h, as detailed in the next section.

$$\bar{\mathcal{P}}_{NET} = h(N_{SF}, \mathcal{E}_{MSG})/T_{MSG} \tag{2.8}$$

---

[2]Radio power provides a mechanism for compensating the excessive consumption of nodes with higher radio traffic. Basically, the distance to the receptor should be lower in areas with more activity, so the transmission power of the device can be decreased without compromising its reliability. Chapter 4 will delve into this topic.
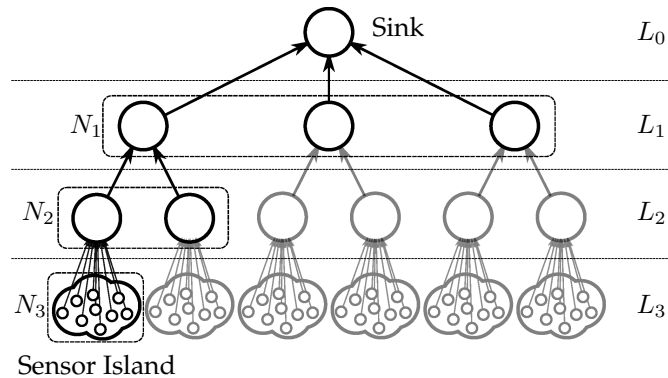
Figure 2.3. *Example of multi-hop WSN scenario.*

Multi-Hop Networks

The above considerations can be generalized to multi-hop networks, where data is routed along some path and eventually collected by a sink node.

A common configuration of multi-hop WSN is shown in Fig. 2.3. Field readings are gathered by the sensors placed at a number of WSN islands (at level L_{K-1}) and then routed to the data collector node (the WSN sink, at level L_0) through a data collection tree. In this scenario, sensors located in the interstitial layers have a dual function. While still generating their local traffic according to some particular distribution, they become repeaters of data coming from the lower layers. This traffic arrives with its own distribution, generated from a physical process that is not necessarily the same, and it may be modified by network activity.

The repeater function of sensors in multi-hop schemes has great impact on a device's total consumption. First, forwarding messages roughly doubles the consumption of merely transmitting (the radio must be switched-on while listening to the incoming message and also while forwarding to the upper layers). Second, since data is aggregated from several child nodes at each layer, the accumulated traffic at each layer grows exponentially. [3]

Time-Synchronized Networks

TSCH networks show a low power consumption profile due to the fact that nodes are synchronized and actions occur at specific moments in *time slots*, enabling nodes to optimize the usage of their resources. Since the actions that occur at each slot are well known, the energy consumption can be modeled on a slot-per-slot basis.

---

[3]Chapter 4 discusses an important application of compression to alleviate this *funnel effect*. Specifically, it turns out that some computational-aggressive compression methods can increase the power consumption locally in the terminal nodes, but still generate important savings in nodes close to the sink, thus balancing energy consumption between members of the network.

Figure 2.4. *Example of two TSCH slotframe configurations with a different number of slots N and M, $M > N$. The first slot is used for network discovery. Then K Data slots for transmission and reception are common in both configurations. Configuration A has $N-K$ Sleep slots (unused), while Configuration B has $M-K$ Sleep slots.*

In a TSCH network, slots are grouped into slotframes, which repeat over time (see Fig. 2.4 and a real capture later in Fig. 2.9a). When a device joins a TSCH network, it obtains information about the duration of each time slot and the number of slots in a slotframe. At each slot, the node can transmit, receive, or keep its radio off. A scheduling entity is responsible for building the schedule, which will satisfy the bandwidth and latency needs of the different flows in the network. The schedule allows for a fine-grained trade-off between latency, bandwidth, redundancy and power consumption.

The overall energy consumption is the sum of the energy consumed in each slot, given that each type of slot has an energy consumption profile related to the hardware and the activity it is performing (e.g., transmit, receive, sleep, etc.). The model used [4] is based on profiling the energy consumption in each of those slots, counting the number of slots of each type and calculating the total aggregated energy of the slotframe.

The average power can be obtained by dividing the aggregated energy E_{SF} by the slotframe period T_{SF}, as indicated by Eq. (2.9). Since slotframes repeat cyclically, this period represents the characteristic temporal scale of the network.

$$\bar{\mathcal{P}}_{NET} = \frac{\mathcal{E}_{SF}}{T_{SF}} = \frac{1}{T_{SF}} \sum_{i=1}^{N_{SLOTS}} \mathcal{E}_{SLOT}^{(i)} \qquad (2.9)$$

In a slot, the actions of different modules (microcontroller, radio, etc.) can be profiled according to the MAC layer timing defined by the standard (e.g., IEEE 802.15.4e) and summarized in [76].

Then, the hardware datasheets can be used to estimate the energy consumed by each action according to the energy state of the components currently being used and the time required by the action.

---

[4]This work adopts the model derived in [76] as a tool for estimating the energy consumption of the addressed scenarios. Note that the aim is to incorporate the previously defined model into the presented methodology. Therefore, this section describes only the indispensable features required for building a higher level system model.

The total energy consumed during a slot is then computed as the sum of the individual contributions, defined mainly by the microcontroller and radio state for each specific action, as shown in Eq. (2.10).

$$\mathcal{E}_{SLOT} = \sum_{k=1}^{N_{ACT}} \mathcal{E}^{(k)}(\mu C, Radio) \tag{2.10}$$

In a TSCH network the slotframe length N_{SLOT} determines how often actions repeat, which usually depends on application requirements. The amount of scheduled cells (i.e.,*Transmit* or *Receive*) depends on the traffic requirements of the application and, as the transceivers are often power-hungry, the communication cost dominates as the traffic on the network increases. Similarly to the spreading-factor discussion; latency, robustness and energy consumption are compromised.

For the purpose of the presented methodology, the key feature is the impact of the network configuration on the energy consumption of the application. Energy consumption can be reduced by increasing the length of the slotframe, i.e., by inserting more *Sleep* slots or by disabling some *Active* slots so that they become *Sleep* slots. A node is only active when in certain timeslots in the slotframe, which are used to send or receive information. In the rest of the non-active slots the node remains switched off. If the number of active slots remains constant and the slotframe size increases, the ratio of sleeping time increases. This means that the average energy spent by the node is smaller, as it is less active. The same effect is obtained by changing sleep slots to active. However, the reduction of activity comes at the cost of less bandwidth and increased latency. Reliability is also compromised, as less redundant links to neighbors are expected.
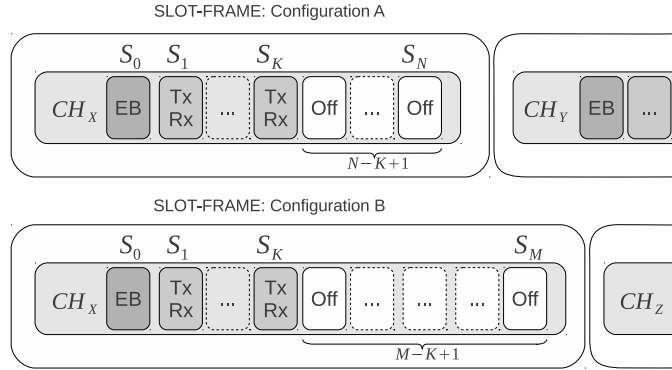
Consider the example given in Fig. 2.4. The diagram shows two TSCH slotframe configurations with a different number of slots N and M, and assuming $M > N$. The first slot is used for network discovery by means of *Enhanced Beacons*. Then K *Data* slots for transmission and reception are common in both configurations. Configuration A has $N - K$ *Sleep* slots (unused), while B has $M - K$ *Sleep* slots, meaning that a node running in this configuration will be idle for longer periods.

2.3.2   Modeling Data Acquisition.

It would be impossible to come up with a model that captures all of the sensing techniques. However it is reasonable to say that most applications fall under the following two categories: regular sensing (with a fixed interval) and event-driven sensing (with some stochastic distribution). In regular sensing, a sensor is woken up at regular intervals to collect one or more samples, and then sent back to sleep. Of course, when energy is freely available, the sensor can be left on permanently; but that is not the case for most battery-operated devices, for which duty-cycling the sensor becomes necessary. In event-driven sensing, a random event triggers a series of samples from the sensor. This event can be internal to the sensor (e.g., sensing activated randomly in *compressed sensing* [28] [16]), or it can be a request for acquired data coming from an external source (e.g., RFID sensors [48]).
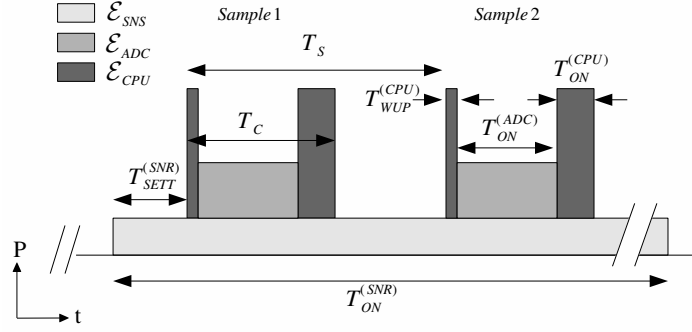
Figure 2.5. *Typical energy consumption $\mathcal{E}_{SMP}$ breakdown for one sample. It is comprised of the contributions from both the sensor and the processor(including the ADC conversion). In this figure T_C is the total capture time; T_S the sampling period; $T_{SETT}^{(SNS)}$ the sensor setting time, i.e., time required for the sensor to stabilize and start capturing; $T_{WP}^{(CPU)}$ is the processor wake-up time; $T_{ON}^{(ADC)}$ the time for the ADC conversion; and finally, $T_{ON}^{(CPU)}$ is the time the processor requires to read a sample from ADC and to store it in memory.*

To quantify the energy consumption of the sensing subsystem in the wireless device, one must first look at the energy required to capture one sample. Then, the energy consumption of the acquisition component is modeled according to Eq. (2.11).

$$\mathcal{E}_{ACQ} = \begin{cases} \mathcal{E}_{SMP} \cdot N_S & (Regular) \\ \mathcal{E}_{SMP} \cdot N_S' \cdot Pr(e) & (Event) \end{cases} \qquad (2.11)$$

In Eq. (2.11), $\mathcal{E}_{SMP}$ is the energy of one sample (see Fig. 2.5), N_S is the number of samples taken during one regular sensing interval, $Pr(e)$ is the probability of an event occurring in one sensing interval, and N_S' is the number of samples taken following the occurrence of that event.

The model can be generalized to Eq. (2.12) in order to account for more than one regular sensing interval (with different periods and sampling requirements) and various event types. In this case, K is the number of sensors sampling on a regular basis, whilst L represents the number of sensors triggered by events.

$$\mathcal{E}_{ACQ} = \mathcal{E}_{SMP} \cdot \left[\sum_{k=1}^{K} N_S^{(k)} + \sum_{l=1}^{L} N_S'^{(l)} \cdot Pr^{(l)}(e) \right] \qquad (2.12)$$

2.3.3 Modeling Local Data Processing Energy

The sensing and networking components are common to all applications running on the same platform and, thus, the same profiling can be reused for further modeling. In contrast, application developers need to be able to estimate the energy consumption for each specific design, sometimes

with limited or even null availability of the actual hardware implementation. Early-stage models should enable the exploration of different alternatives while minimizing the risk.

A method for estimating the energy required by a software task was proposed and validated originally in [J2]. Starting from a high level description of the algorithm (e.g., Matlab/Octave), the number of operations to process the original sensed signal is recorded, accounting basically for the number of arithmetic operations: additions, multiplications, divisions and comparisons –i.e., all the main actors in signal processing loops. Thus, depending on the selected hardware architecture, these counters are mapped into the corresponding number of microcontroller clock cycles, and the latter is subsequently mapped into the corresponding energy expenditure. Please refer to [J2] for further details.

This method offers accurate results as long as the CPU tasks rely mainly on arithmetic instructions (as digital signal processing algorithms do). However, it is no longer applicable when the microcontroller is involved in non-arithmetic based tasks, like dealing with a protocol stack. In that case, alternative methods based on ISS simulators can be used [9].

2.4 Fitting Technological Parameters

For the sake of analytical tractability, it is often convenient to make an intermediate fitting step to find a simple closed-form expression of each of the individual contributions. As this step requires experimental measurements of the actual platform, it can be skipped in early stage development (e.g., when the platform is not yet available) or if the required time and resources do not justify the additional benefit. Even so, as it is based on empirical measurements, fitting can significantly improve the accuracy of the model, and so it is strongly recommended. The following set of examples illustrates the fitting procedure for different types of components.

2.4.1 Units and Measurements

Before starting with measurements, it is worth noting a few considerations. First, one measures current, not power. An appropriate conversion must be applied to use all the above formulations in terms of current. The actual power drained from the buffer is computed according to $\mathcal{P}_{DEV}=I_{DEV}\cdot V_{BUF}$, where I_{DEV} is defined as the current measured at the output of the energy buffer and V_{BUF} the voltage in its terminals (See Fig. 2.1) Moreover, the reference values in the datasheets of the components (chips, batteries, scavengers, etc.) are typically expressed in intensity units. But power on the components side is relative to the local voltage. Then, for each individual component i connected to the power domain j, the power is obtained according to: $\mathcal{P}_{CMP}^{(i,j)}=I_{CMP}^{(i)}V^{(j)}$.

Obviously all magnitudes must be compared in the same domain. As measurements are easier at the buffer output, before current reaches the regulators or DC/DC converters, it is recommended to operate on the buffer side. In fact, for any practical estimation, what actually really matters is the load from the energy buffer.

The class of regulator determines how to jump from one domain to another. DC/DC converters basically preserve power (with some losses being parametrized with their efficiency factor η). Then, to interpret component currents as battery loads, the proper conversion is given by:

$$DC/DC: \; \mathcal{P}_{OUT}=\eta\mathcal{P}_{IN} \Rightarrow I_{DEV}^{(i)} = I_{CMP}^{(i)}\frac{1}{\eta}\left(\frac{V^{(j)}}{V_{BUF}}\right)$$

Linear regulators roughly preserve currents, provided that the minimum required voltage dropout δ is respected ($V_{OUT}>V_{IN} + \delta$). Then, currents measured on the buffer side and the actual currents on the device side are approximately the same:

$$Linear: \; I_{OUT} \cong I_{IN} \Rightarrow I_{DEV}^{(i)} \cong I_{CMP}^{(i)} \;\; \forall V^{(j)} \mid V_{BUF} > V^{(j)} + \delta$$

While respecting these rules, models can be described in current units instead of power, and charge units instead of energy (i.e., normalized by the voltage). This allows avoiding continuous conversions and facilitates experimental measurements. When all the device's components share the same power domain, the conversion is almost direct.

2.4.2   Sampling Characterization

<div align="center">

(a) *Superimposed set of samples.*   **(b)** *Empirical charge distribution.*
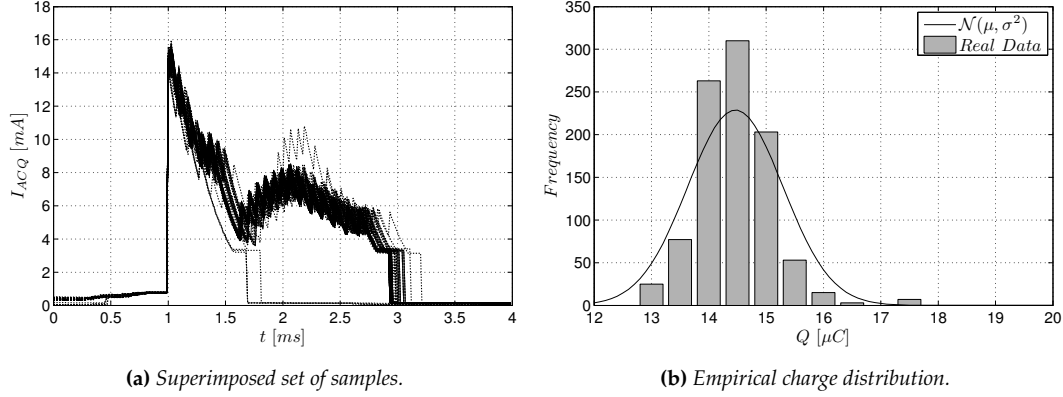
Figure 2.6. *Characterization of sensing charge. Fitting to a normal distribution.*

</div>

To compute the average current of the acquisition block, the charge drawn to get the N_S samples that form a record is divided by the time elapsed between consecutive records T_{RCD} (i.e., the wake-up period), following Eq. (2.13). In this approximation, $\bar{I}_Q$ accounts for the stand-by or quiescent current of the sensor, while $\bar{Q}_{SNR}$ can be interpreted as the average charge necessary to get one sample. It is comprised of the contributions from both the sensor and the processor (including the ADC conversion), which are represented in Fig. 2.5.

$$\bar{I}_{ACQ} \cong \frac{\bar{Q}_{SNR} \cdot N_S}{T_{RCD}} + \bar{I}_Q \tag{2.13}$$

The value of $\bar{Q}_{SNR}$ can be obtained by fitting a set of experimental samples. To illustrate the procedure, Fig. 2.6 summarizes an experiment performed to characterize a digital magnetometer sensor. Fig. 2.6a shows the current measured during the acquisition process, with multiple samples superimposed to portray the variability between them. The charge of each individual sample is obtained by integrating the measured current. Fig. 2.6b shows the empirical distribution of a dataset of ≈ 1000 samples. The average charge per sample can be approximated to the mean value of the distribution $\bar{Q}_{SNR} \approx \mu_Q$. In this case, data is fitted to a normal distribution with $\mu_Q \pm \sigma_Q = 0.0145 \pm 0.0008 \, [mC]$.

2.13 makes explicit the impact of duty-cycling in the consumption. While in many situations the sampling period T_S is bound by the underlying physical magnitude and filtering requirements (e.g., AC noise filtering in magnetic readings), the time between consecutive records T_{RCD} is scheduled from the application layer, thus providing a mechanism for adjusting the trade-off between energy consumption and sensing accuracy. In the next sections, several examples of this trade-off are discussed.

2.4.3   Network Profiling

To devise a useful model, it is important to figure out the functional dependence on some control parameters. The procedure is in essence the same for many types of networks. First, one should identify a suitable parameter, N_X, to gain control over the power consumption. Then, it is necessary to fit the experimental data to the analytical function or polynomial approximation $\mathcal{H}(N_X)$ in Eq. (2.14), which is chosen to model the functional dependency on N_X (See Eq. (2.8)). The following examples illustrate the procedure for two different networks.

$$\bar{Q}_{NET} \cong \mathcal{H}(N_X) + \bar{Q}_B \tag{2.14}$$

Long-Range Communications

Semtech is a provider of low-power, wide area, ISM band radio, based on a proprietary protocol called LoRa **??**. In the LoRa approach, base-stations monitor several bands and devices, which are synchronized, and they can send packets in their own time-slot and wait for the ACK of every packet. These ACK packets can embed some feedback information, as well as notifications of an incoming downlink packet. To increase the range, LoRa uses a configurable *SpreadingFactor* (SF), i.e., the ratio between clock rate and the symbol rate. This SF parameter can be configured from SF6 to SF12 (64 to 4096 chips/symbol), with an increase in the link budget of 14 dB in the highest SF. This ends with a reduction in bit-ratio, which affects the time needed to send a payload, and hence the energy consumed in each transmission. Thus, the spreading-factor is used to extend communication range or improve immunity to interference, although at the expense of more energy consumption for each message transmitted.

Fig. 2.7 shows the device current, as measured for radio transmissions with different spreading factors. In this experiment, the protocol stack is configured with up to 3 retransmissions if message delivery fails. For illustrative purposes, the gateway is switched off. Consequently, we can see in these records the three transmission attempts (pulses with maximum current) and the respective waiting times for confirmation (small pulses), when radio is switched on to listen for the ACK messages.

The effect due to the different modulation is noticeable in the figures. Basically, each step in the spreading-factor scale doubles the time the radio spends in its active state. This suggests that the charge per message can be parametrized with an exponential function $\mathcal{H}(N)$ in the form $\mathcal{O}(h(N))=2^N$, leading to Eq. (2.15) as a tentative fitting function. In this case, it can be interpreted as the charge in the basic configuration.

$$\bar{Q}_{NET} \cong \bar{\mathcal{Q}}_{MSG} \cdot 2^{N_{SF}} + \bar{Q}_B \tag{2.15}$$
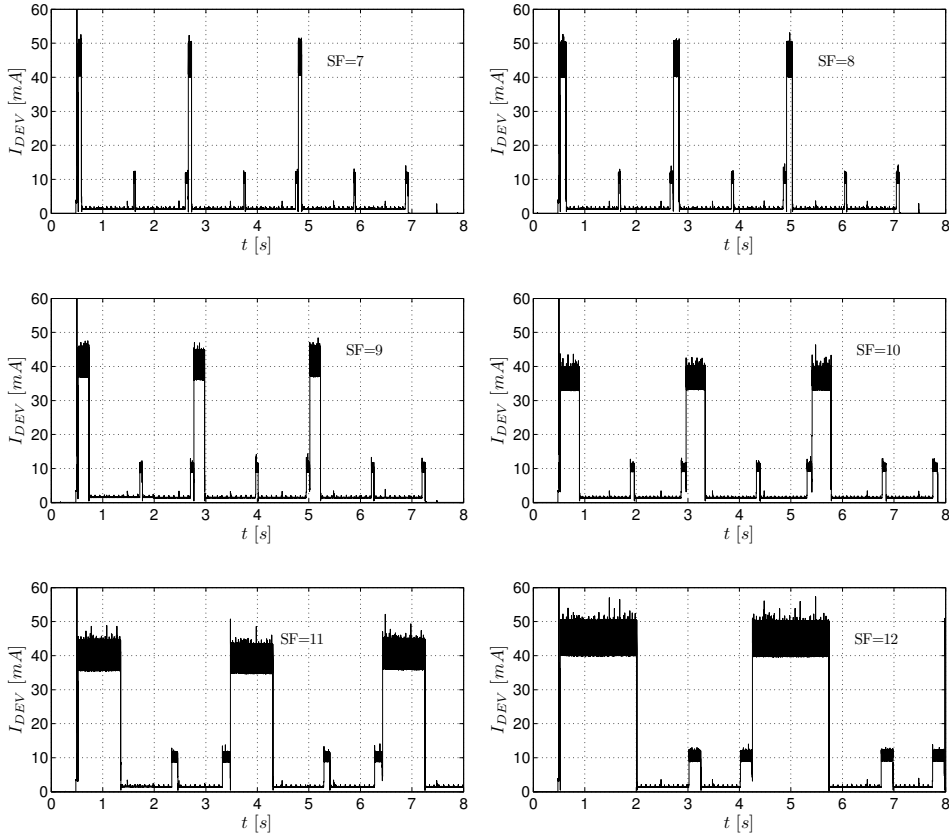
Figure 2.7. *Characterization of Cycleo transmissions using different spreading-factors*
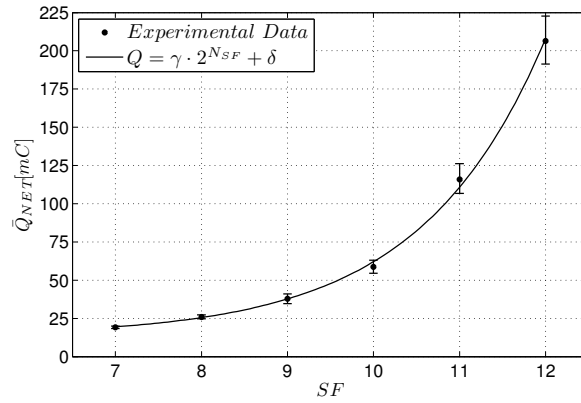
Figure 2.8. *Fitting Cycleo spreading-factor to Eq. (2.15).*

Fig. 2.8 shows the model fitted with a training set of ≈ 100 samples for each modulation. Results demonstrate that the postulated model is in full agreement with real measurements, always within the experimental error. Numerical values obtained are $Q_{MSG} = 6.1\,\mu C$ and $\bar{Q}_B = 13.0\,\mu C$.

TSCH Networks

(a) *Slot-frame composed by 10 slots with 5 active.*

(b) *Fitting N_{SLOTS} parameter to Eq. (2.16).*
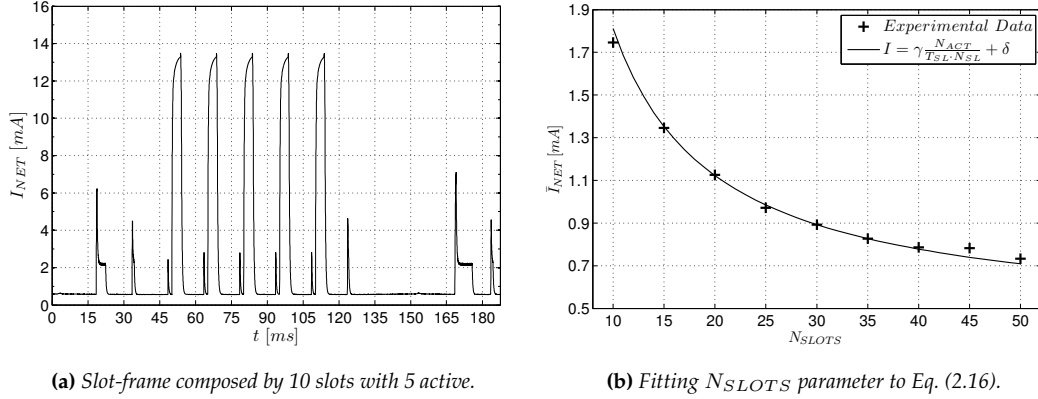
Figure 2.9. *Characterization of TSCH radio.*

To profile the energy consumed by TSCH networks, Eq. (2.9) should be computed to determine the charge used in each slotframe with different network setups. Details of the procedure can be found in [76]. As a complementary approach, the average current required to maintain the network can be empirically approximated.

To obtain a suitable fitting function, recall from Eq. (2.9) that the average current can be obtained from the ratio between the total charge of the slotframe Q_{SF} and the period of the slotframe T_{SF}. The charge can be roughly estimated based on the number of active slots and the charge per active slot $Q_{SF} \approx Q_{MSG} \cdot N_{ACT}$, whilst the length of the slotframe is determined by the total number of slots and the duration of each slot $T_{SF} = T_{SLOT} \cdot N_{SLOTS}$ [75]. With these assumptions, Eq. (2.16) should be a reasonable approximation for the average current. Here, $\bar{I}_B$ represents the background activity of the μC to control the network (periodic wake-ups, synchronization messages, etc), and it can be considered constant. Fig. 2.9b shows the empirical fitting of Eq. (2.16), which is obtained for a GINA platform [49]. The numerical parameters can be found in [J3].

$$\bar{I}_{NET} \cong \frac{\bar{Q}_{MSG} \cdot N_{ACT}}{T_{SLOT} \cdot N_{SLOTS}} + \bar{I}_B \tag{2.16}$$

In Eq. (2.16), the duration of a timeslot T_{SLOT} is a fixed network parameter. $\bar{Q}_{MSG}$ represents the average charge required per packet and depends basically on the radio technology. This means that, once the number of active slots N_{ACT} in the slotframe are defined, the number of slots N_{SLOTS} in the frame becomes the control parameter for the network energy, giving a characteristic functional dependency $I_{NET} \propto 1/N_{SLOTS}$ that is obtained empirically in Fig. 2.9. Eq. (2.16) shows explicitly the impact of the duty-cycle, parametrized in this case by the period of the slotframe through N_{SLOTS}.

(a) *Processor current for different number of points.* (b) *Accumulated charge for different run lengths.*
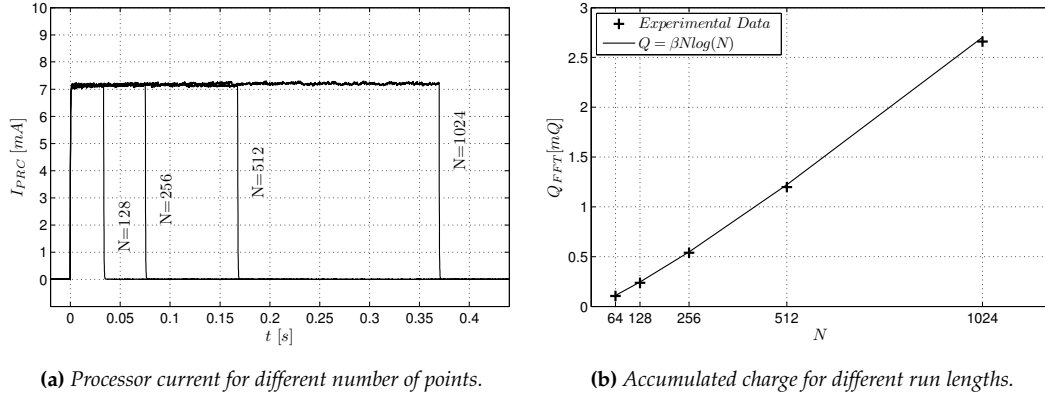
Figure 2.10. *Fitting parameters of $\mathcal{FFT}$ computing.*

2.4.4   Processing

As outlined in Sec. 2.3.3, a high level implementation of a given algorithm can be used to make a first estimate of the computation time (more details about this method can be found in [88]). Alternatively, the code can be ported to a device's specific implementation and simulated with an Instruction Set Simulator.

Table 2.1 compares both approaches with real measurements obtained from a custom implementation of the FFT algorithm (Matlab/C). Results show that the time estimated through the number of MATLAB operations $\widehat{T}_{MAT}$ did not differ significantly from that derived with the simulator $\widehat{T}_{ISS}$, and the latter is in perfect agreement with the time measured using the algorithm running on the MSP430 processor T_{PRC}, shown in Fig. 2.10a

Table 2.1. *FFT algorithm processing time.*

N_{FFT}	$T_{PRC}[ms]$	$\hat{T}_{MAT}[ms]$	$\delta T_{MAT}[\%]$	$\hat{T}_{ISS}[ms]$	$\delta T_{ISS}[\%]$
64	14.6	14.3	2.1	14.7	0.4
128	33.5	33.2	0.8	33.5	0.0
256	75.3	75.8	0.6	75.4	0.1
512	167.8	170.2	1.4	168.0	0.1
1024	369.8	377.8	2.1	370.3	0.1
	Measured	*Matlab*		*I.S. Simulator*	

To come up with a valid model of the algorithm, it is important to identify the functional behavior that depends on one or more control parameters. For already known algorithms, a natural choice is the (worst-case) time complexity $\mathcal{T}(n)$. On the other hand, if the algorithm is custom designed or its complexity is unknown, the fitting function can be a generic series expansion when characterizing compression algorithms, as discussed later in Chapter 4.

To illustrate this former idea, we use the *Fast Fourier Transform* (FFT) as a case-study algorithm. The FFT implementation of the DCT has a well known $\mathcal{T}(N){=}\mathcal{O}(f(N)){=}N\,log(N)$ complexity. Accordingly, the associated processing time should be proportional to this $Nlog(N)$ relation. Considering $\bar{Q}_{OP}$ to be an estimate of the average cost per arithmetic operation, Eq. (2.17) can fit processing power consumption with reasonable accuracy, as shown in Fig. 2.10b.

$$\bar{I}_{PRC} \cong \frac{k\,\bar{Q}_{OP} \cdot N\,log(N)}{T_{RCD}} + \bar{I}_{SYS} \tag{2.17}$$

In Eq. (2.17), k is a constant factor that depends on the algorithm's implementation. In this case, it is related to the number of arithmetic operations per FFT point. $\bar{I}_{SYS}$ includes all system related functionalities of the μC, such as running the operating system, managing periodic interrupts, etc.

2.5   System Models

2.5.1   Putting the Pieces Together

The next step in the modeling process consists of merging all contributions into one single expression. Continuing with the examples Eq. (2.18) combines the contributions discussed in sections 2.4.2, 2.4.3 and 2.4.4, keeping technological and application parameters as independent variables.

Constants α, β, γ and δ depend only on the particular sensor, MCU and radio technologies. Recalling the meaning of each individual contribution from the fitting process examples, α can be interpreted as the charge per sample $\bar{Q}_S$; β represents the cost per operation $\bar{Q}_{OP}$ associated with the specific μC; while γ is an estimator of the average charge per message $\bar{Q}_{MSG}$. Arranged in this way, Eq. (2.18) allows a straightforward evaluation of alternative technologies by merely specifying suitable values for these parameters.

In turn, T_{MSG}, N_i and T_{RCD} are application parameters that may be adjusted in order to meet the specifications once the specific technology has been established. Recalling Eq. (2.18) from the previous section, N_S stands for the number of samples effectively acquired in each sampling interval; N_A parametrizes the amount of data to be processed; and N_X represents any parameter related to radio activity. T_{MSG} and T_{RCD} weigh the duty-cycle and have a fundamental impact because of the $(1/x)$ functional dependence.

$$\bar{I}_{DEV} = \frac{\alpha N_S}{T_{RCD}} + \frac{\beta \cdot \mathcal{T}(N_A)}{T_{RCD}} + \frac{\gamma \cdot \mathcal{H}(N_X)}{T_{MSG}} + \delta \tag{2.18}$$

This is a good moment for us to look back briefly and recapitulate. Starting from a vague condition asserted in Eq. (2.2), the benefits drawn from the methodology introduced in this section depend on being able to make Eq. (2.18) tangible. First, this general expression should be properly interpreted according to the specific platform (technological parameters) and application (operational parameters). The outcome naturally emerges when exploiting by simulation the analytical model, i.e., the particular instance derived from Eq. (2.18).

The next sections illustrate this procedure in further detail by means of two case-studies based on real applications. Subsequently, Chapter 3 carries out a comprehensive analysis under this formulation of self-powered devices (an outstanding topic of current research) and, thus, demonstrates the effective application of this methodology.

2.5.2 Case Study I: Periodic Reporting Applications

Network Scenario

As mentioned previously, radio transceivers are evolving towards modulation techniques with wide spread spectrum. These modulation techniques provide ultra-long range communications while maintaining low energy consumption, thus being suitable for battery powered devices.

A notable example of this technology is Weightless [84], an industry consortium originally founded by the UK company NEUL with more than 1,000 members operating under some specific license agreement. Weightless fosters the development of wide-area communication in white spaces, covering ranges of up to 10km. Another example of wide-range wireless connectivity for M2M is Sigfox [66]. Sigfox uses a simple radio technology referred to as Ultra Narrow Band (UNB) and operates in the license-free ISM frequency band of 868 MHz. Cycleo (now Semtech) is another provider of low-power, wide area equipment operating in the sub GHz ISM bands, which is based on their proprietary approach called LoRa [65].

All these technologies operate over distances great enough to avoid multi-hop techniques. The wide range, combined with low data-rate orientation, allows modeling the radio activity of these devices based on the following assumptions: point-to-point, unidirectional link and retransmission free. All of these are characterized in Section 2.3.1.

Application Examples
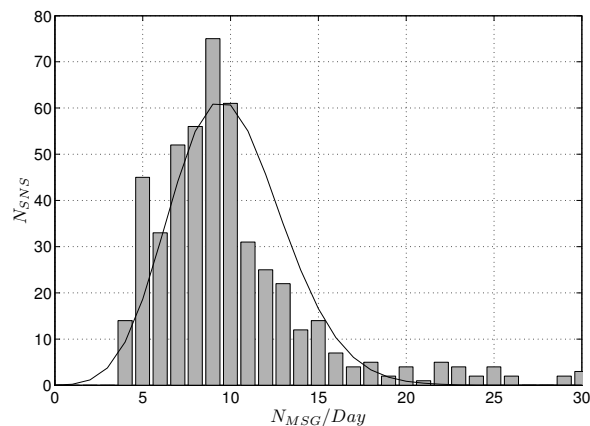
Figure 2.11. *Traffic generated by ≈ 500 parking spot sensors. Data collected over 100 days.*

Despite the simplicity of this scenario, it covers a large number of applications. Spread spectrum technologies are becoming true enablers of the IoT by means of the huge range of applications that will appear thanks to the long-range paradigm. It is probable that the most representative application of the wide-area, low bit-rate, low-cost approach is the huge smart-metering market.

Smart parking systems are another good example for this case study. On-street parking sensors are small devices used to monitor the availability of individual parking spots. Each device periodically wakes-up to check the state of the spot. When a car parks over it, its presence is detected, and the sensor relays the event to the gateway. The bandwidth required for this application is particularly low. The typical interval time between radio messages oscillates between some minutes and several hours, and the information required per message is very small, as the state can be codified with 1 bit. For this reason, the smart-parking application is particularly well suited for long-range radio technologies.

Data generated from a single parking sensor is basically unpredictable, as it depends on multiple pseudo-random human factors. However, when data coming from a set of similar sensors is aggregated, a typical Poisson-like distribution emerges. The expected value of this multiple-sensor spatial-distribution can be used as an estimator of the temporal-distribution of a single sensor [5].

Fig. 2.11 shows an example of distribution found in a real parking application. The histogram was obtained from the events gathered by a set of sensors belonging to the same sector and recorded over several days. The solid line represents the fitted Poisson distribution. In this case, the expected value of the number of messages per day is given by the Poisson mean $E[N_{MSG}] = \lambda_{MSG}$, and it is used to estimate the average lapse time between messages $\widehat{T}_{MSG} = 1/\lambda_{MSG}$, as indicated in Eq. (2.8).

Model description
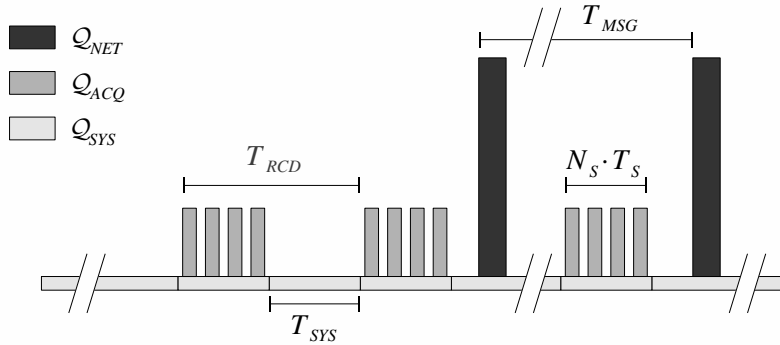
Figure 2.12. *Characteristic time evolution of a simple reporting application.*

Embedded applications are defined by their repetitive operation, and the first step in modeling them is to find the patterns of repetition. In this case, a *record* of N_S samples is acquired with a fixed interval time T_{RCD}, while some information is reported to the data collection center with some characteristic period $\widehat{T}_{MSG}$ (see Fig. 2.12). In this simple approach, the reported information is aggregated in a unique message, and this message is retransmitted a certain number of times, N_R, to

---

[5]Chapter 5 presents a detailed study of this subject.

increase the probability of success. It thus follows the SIGFOX approach as an alternative to implementing an acknowledgment scheme on the downlink of LoRa. In this approach, the functional dependence associated with network retransmission is trivial: $\mathcal{H}(N_R){=}N_R$. In this case, γ is easily understood as the charge per message $\bar{Q}_{MSG}$.

Usually, the number of samples acquired (N_S) is the same as the number of samples processed (N_P). This means that the parameter simultaneously affects the energy contributions of both sensing and processing tasks. However, being that this particular case is just a reporting application, the cost associated with processing is very low, and the β term can be omitted. Then, Eq. (2.19) combines Eq. (2.6) and Eq. (2.13) in a basic instance of Eq. (2.18).

$$\bar{I}_{DEV} = \frac{\alpha N}{T_{RCD}} + \frac{\gamma N_R}{T_{MSG}} + \delta \tag{2.19}$$

Even more, all data acquired is transmitted for some basic reporting applications. Consequently, the time between records T_{RCD} and the time between messages T_{MSG} become equal –merging the first and second terms in Eq. (2.19). This wake-up period simultaneously modulates sampling and communication contributions, and can therefore be considered a global control parameter. This model is particularly important for opportunistic approaches such as [71].

Simulation

Fig. 2.13 presents a simulation obtained by applying Eq. (2.19) to different sampling and recording period configurations, using the specific technological values for LoRa and the magnetic sensor obtained in Sections 2.4.2 and 2.4.3 (See also Fig. 2.6 and Fig. 2.8). The bars show the contribution to the overall energy of the network and sampling components, according to the time elapsed between consecutive messages and the sampling rate. The floor level is associated with system management (e.g., periodic interrupts of the operating system). Although it is constant, this contribution carries important weight in this application. The reason for this is that radio activity is very low and its contribution is not dominant, as is often considered in the literature. Processing cost is also represented, despite being negligible for this particular application.

Asymptotic behavior can be seen in both axes of Fig. 2.13. This is a consequence of the $(1/x)$ functional dependence of the duty-cycle. If the recording interval is fixed, the overall energy consumption is reduced when radio activity is lower. However, the amount of energy that can be saved is limited by asymptotic decreasing. At a certain point, increasing the interval between messages does not significantly reduce the energy consumed. Analogously, as the recording interval increases, the energy savings decrease.

This graphical representation is useful as a tool for determining which of the control parameters
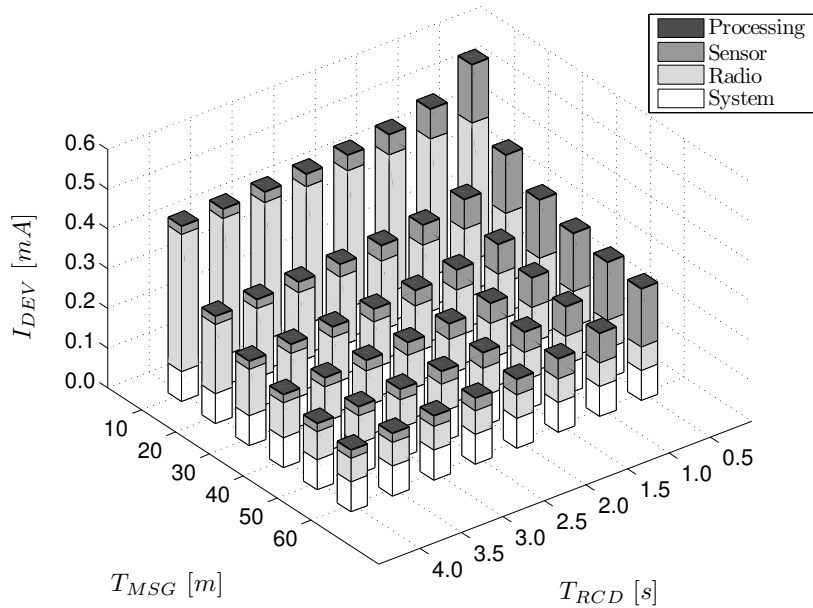
Figure 2.13. *Simulated consumption of a simple periodic reporting application.*

yields the highest energy savings, as well as for figuring out the achievable gains and limits for optimization.

Validation

A set of experiments have been carried out to validate the model. The employed platform was composed of a Cortex-M4 32 bit μC, a Telecom Designs TD1202 long-range radio module and a Honeway HMC5883 compass IC. The RTOS wakes-up periodically with a systick interrupt of 1 ms.

The application was configured for different sampling intervals T_{RCD} and different reporting period T_{MSG}. Fig. 2.14 compares the experimental results with those predicted by Eq. (2.19). The plus sign markers in the figures are the actual measured currents, while the vertical bars show the estimated values. Gray areas account for the statistical deviation in the fitting process of the technological parameters.

As shown in Fig. 2.14, all estimated values lie within the statistical error. These results demonstrate that, despite all assumptions made, the accuracy of Eq. (2.13) is still rather acceptable. In fact, the source of uncertainty is not the systematic error inherent to any model, but errors introduced in the characterization of individual tasks. These errors can be caused by multiple pseudo-random factors (such as chip to chip variations in production, changes in the temperature of operation or any other environmental issues), and therefore must always be taken into account for worst-case predictions.
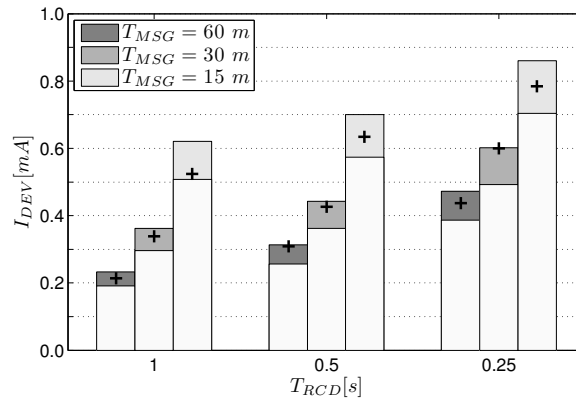
Figure 2.14. *Model validation of a simple periodic reporting application. The markers are the actual measured currents, vertical bars show the estimated values and the gray areas account for the statistical deviation in the fitting process of the technological parameters*

2.5.3 Case study II: TSCH Networks

Network Scenario

Industrial Wireless Mesh networks are being consolidated by standardization efforts under the Time Slotted Channel Hopping (TSCH) scheme. This technique has been adopted by major industrial, low-power wireless standards such as WirelessHART [1], ISA100.11a [35] and, more recently, as a part of the IEEE802.15.4e standard [2]. As of today, several commercial low-power wireless networking providers are offering almost 100% reliable MAC layers [27], while at the same time providing radio duty cycles well below 1%, thereby reducing the power consumption and increasing network lifetime. This is facilitating the introduction of new monitoring and actuating devices that aim to improve the security, process automation, efficiency and productivity of the industries. Furthermore, it devises a clear roadmap for the Industrial Internet paradigm. Nowadays, industrial wireless communication is considered a mature technology.

Application Examples

Multiple industrial monitoring applications can fit into this category by taking advantage of the reliability of TSCH networks. Notable examples include vibrational analysis of rotary machines, structural health monitoring through harmonic analysis, accelerometers for monitoring power-line towers [86], and vibrating wire strain gauges [14] for measuring infrastructures.

Additionally, all these examples involve some kind of frequency analysis that can be performed by means of the FFT algorithm, which has been chosen for illustrative purposes in this case-study. Chapter 3 makes use of this model for two representative applications.

Model Description
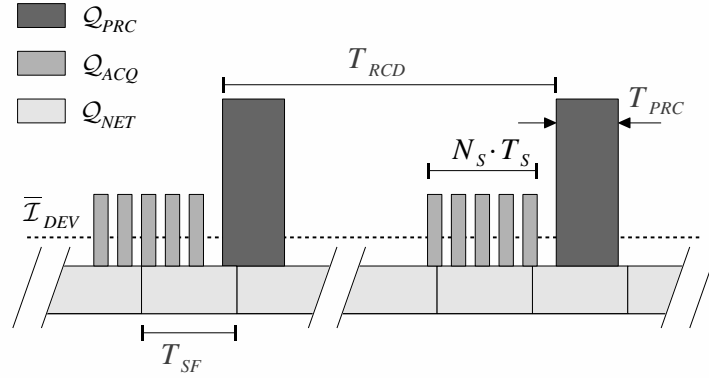
Figure 2.15. *Characteristic time evolution of a device associated with a TSCH network. The system wakes up with T_{RCD} period. Each cycle, it acquires a set of N_S samples with sampling period T_S and requires T_{PRC} to process it. In the background, TSCH radio maintains network synchronization and reports health of the system within T_{SF} cycles.*

This second example models an application that requires some arithmetic computing. On top, the processor manages the link to a TSCH network, requiring some communication activity in order to maintain synchronization with the network. This network management is handled in background within slot-frames. Fig. 2.15 shows the operational cycle of this type of application. In this figure, the network's periodic management is carried out in T_{SF} cycles.

More formally, Eq. (2.20) combines the contributions of (2.13), (2.16) and (2.17), which account for, respectively: the acquisition, processing and TSCH network management tasks.

$$\bar{I}_{DEV} = \frac{\alpha N}{T_{RCD}} + \frac{\beta N log(N)}{T_{RCD}} + \frac{\gamma N_{ACT}}{T_{SLOT} N_{SLOTS}} + \delta \qquad (2.20)$$

The main parameter involved in network consumption is N_{SLOTS}, which is related to the number of *Active* and *Sleep* slots.

Typically, the number of points computed by the FFT N_P and the number of samples read by the ADC N_S are the same. So, as in the previous case, this parameter simultaneously affects both contributions and is denoted simply by N. With N fixed, the duty-cycled behavior of the application makes the time between records T_{RCD} the fundamental parameter for controlling the average power. Specifically, as the time between records increases, less power is consumed. Therefore, the time interval between consecutive records determines the time scale for power averaging.
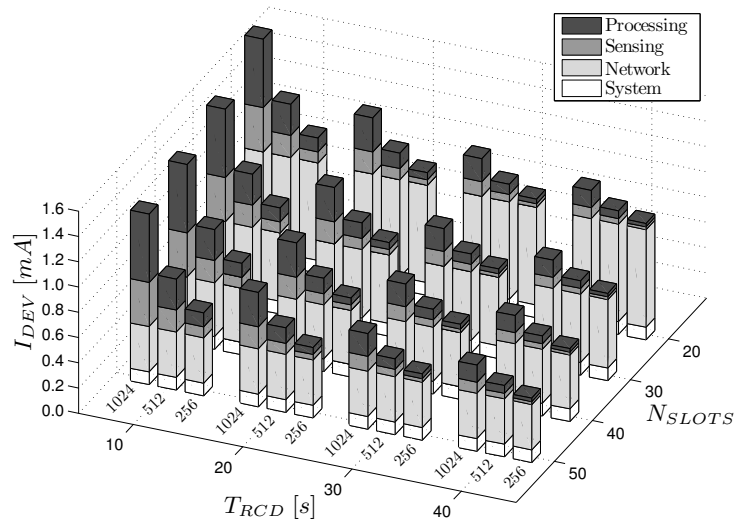
Figure 2.16. *Simulated consumption of an application using TSCH network*

Simulation

Fig. 2.16 presents a simulation obtained by applying Eq. (2.20) to different network and recording period configurations, considering 256, 512 and 1024 samples per record. The bars indicate the contribution to the energy consumption of the network, sampling and processing components, according to the number of slots per slotframe and the recording interval.

Again, asymptotic behavior appears in both axes of Fig. 2.16. While maintaining a fixed interval time, the overall energy consumption is reduced by increasing the number of slots in a slotframe. Still, the asymptotic behavior sets a limit to the power savings. At a certain point, increasing the number of slots in the network does not significantly reduce the energy consumed.

Validation

Experiments were carried out using a GINA platform [49] and running the OpenWSN protocol stack [83]. The GINA platform comprises several inertial sensors for angular rate and linear acceleration, along with a general purpose microprocessor. Specifically, constants α, β, γ and δ of Eq. (2.20) have been characterized for the Texas Instruments MSP430f2618 16-bit μC, the Atmel AT86RF231 IEEE802.15.4 radio, and the ST-LIS344ALHTR 3-axis accelerometer sensor.

Fig. 2.17 compares the experimental results with those predicted by Eq. (2.20) for different application configurations. The plus sign markers in the figures represent the measured current values. The vertical bars represent the values estimated by the model, showing the uncertainty introduced by the fitting process.

(a) $N_{FFT} = 1024$

(b) $N_{FFT} = 512$

(c) $N_{FFT} = 256$

(d) $N_{FFT} = 128$
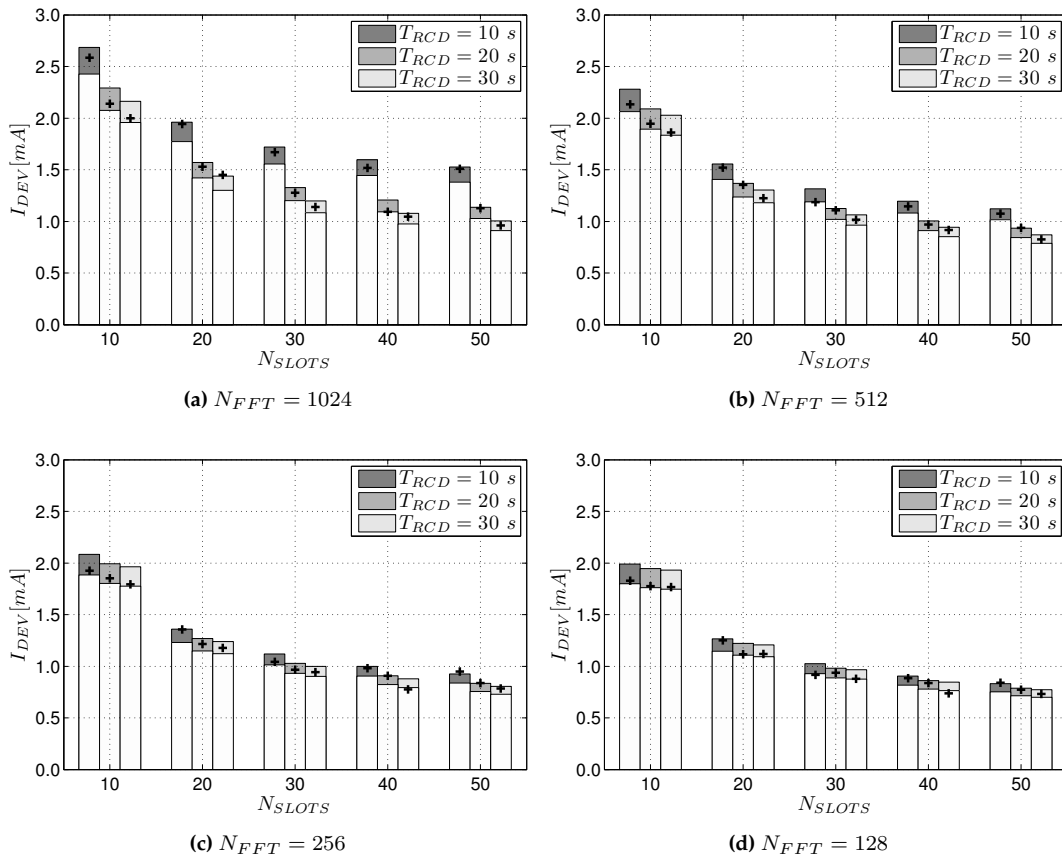
Figure 2.17. *Comparison of model predictions and experimental results for different parameter configurations. The plus sign markers in the figures represent the measured current values. The vertical bars represent the values estimated by the model, showing the uncertainty introduced by the fitting process.*

Discussion

The present chapter improves on the current state of the art by defining a general methodology and introducing a new perspective on the energy characterization and dimensioning of low-power, embedded devices. Using these models, which can be applied to different industrial applications, engineers are able to foresee the behavior of the different application parameters and their impact on power consumption, even without a complete implementation of the application. Hence, using this framework can help engineers study the viability of a new application in terms of power consumption, energy harvesting needs, battery requirements, etc.

Experimental measurements of a system provide only the energy consumed by the entire system without adding any knowledge about the distribution of consumption throughout the different subsystems. This makes it difficult to identify the main contributors to the energy consumption and the parameters upon which the application depends, thereby limiting the scope of energy optimization. The presented model, however, allows to determine the amount of energy consumed by each of the main subsystems: standard networking technologies, sensing and processing. In this way, it makes possible to understand the energy balance between them. The results demonstrate that the contribution of the different components is of the same order of magnitude; and, consequently, all of them must be taken into account in order to explore the feasible options and fine-tune the application parameters. This systematic approach has not been tackled by the current literature.

The presented experimental results demonstrate the model's suitability for different scenarios and platforms. The validation has been performed using two different applications. After fitting the technological parameters and comparing them with real measurements, excellent simulation results were achieved.

Finally, it is important to establish the scope and applicability of the model. A device operating in a real-world scenario always has a degree of uncertainty associated with environmental conditions. Variations in the PDR are just a representative example, but there are many other issues that can significantly affect the functionality of the system and therefore its energy consumption (e.g., temporal link interruptions, unexpected system restarts, etc.). Under these circumstances, further refinements for improving the accuracy of estimates may not be necessary. Instead, the strength of the model lies in its capability to support better-informed decisions and avoid risks in the early stages of development.

The full potential of this methodology will be demonstrated throughout the remaining chapters by applying it to three different hot topics on low-power research.

Chapter 3

Self-Powered Sensor Networks

The use of renewable energy is desirable at every level of society, from industrial and manufacturing activities to smart cities, public buildings, etc. In particular, the ability to capture any sort of renewable energy source is an appealing prospect for powering up the sensing equipment and electronic devices that surround us in our daily life, whether they be automatic doors, sensor systems for traffic control, intrusion detectors, alarms or pollution monitors, among many other possibilities.

After having been restricted to scientific research only a few years ago, energy harvesting in small form-factor embedded systems is now becoming an essential technology in the field of autonomous wireless networked systems. Of course, the ultimate desire is to cut down their maintenance costs with self-sufficient networks that could operate unattended, thanks to the energy they extract from the environment. Yet, the road to perpetual wireless communications devices is still full of challenges: ambient energy is intermittent and scarce, energy storage capacity is limited, and devices are constrained in size and complexity.

With the promise of self-sustainable, maintenance-free networks comes a fundamental shift in design principles, especially when compared to traditional battery-operated devices: whereas minimizing energy consumption is crucial to extending the lifetime of the latter, the objective for self-powered devices is to manage the harvested energy intelligently in order to ensure long-term, reliable and uninterrupted operation.

3.1 Introduction: Principles of Self-Powered Operation

3.1.1 Energy Neutral Design
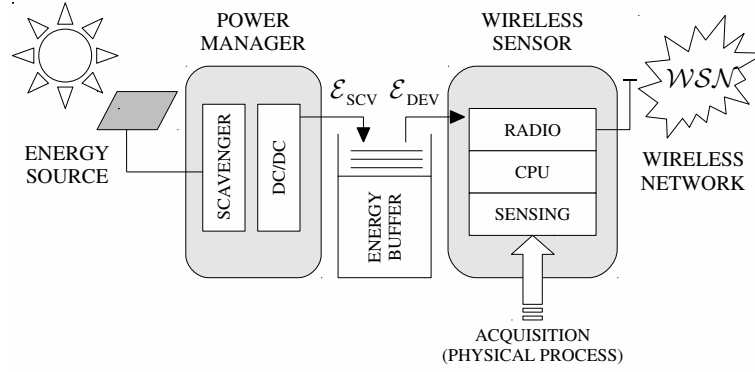
Figure 3.1. *Energy harvesting wireless sensor device.*

A SELF-POWERED, wireless sensor device is characterized by a sustainable provision of energy. In the general model presented in Fig. 3.1, the energy scavenged from the medium $\mathcal{E}_{SCV}(t)$ must be greater than the energy required by the device $\mathcal{E}_{DEV}(t)$, in accordance with Eq. (3.1) [1]. If this condition arises, the energy income and outcome are, at least, compensated. This concept is commonly referred to in the literature as *energy neutral design* [15].

$$\mathcal{E}_{SCV}(t) = \int_0^t \mathcal{P}_{SCV}(\tau)d\tau \geq \int_0^t \mathcal{P}_{DEV}(\tau)d\tau = \mathcal{E}_{DEV}(t) \tag{3.1}$$

A natural constraint follows from *energy causality*, which dictates that energy cannot be used before it is harvested [34, 72]. This is equivalent to keeping the expended energy curve (accumulated) under the harvested curve the whole time (See Fig. 3.2). Formally, causality requires that the Eq. (3.1) condition must hold for any value of t, as indicated in Eq. (3.2).

$$\mathcal{E}_{SCV}(t) \geq \mathcal{E}_{DEV}(t), \quad \forall t \tag{3.2}$$

Due to cost and mechanical constraints, it is also a desirable condition the harvested energy not be oversized, or equivalently, that all the harvested energy should be used by the deadline. That is, the transmitted energy curve should converge to the harvested curve over the long term in Fig. 3.2. Eq. (3.3) formalizes this condition:

---

[1]Eq. (3.1) emerges from Eq. (2.1) when considering null the initial buffered energy; e.g., when there is no battery at all, or when the time elapsed from start-up allows us to consider the effect of the initial contents negligible.

$$\mathcal{E}_{SCV}(t \to \infty) \approx \mathcal{E}_{DEV}(t \to \infty) \tag{3.3}$$

Moreover, due to the finite buffer size, any received energy that overflows its capacity is lost, causing an undesirable waste. The amount of energy buffered at any instant t is basically the difference between the harvested energy curve and the demanded energy curve in Fig. 3.2. This difference should never be larger than the buffer capacity [72], a condition that is reflected in Eq. (3.4).

$$\mathcal{E}_{SCV}(t) \leq \mathcal{E}_{DEV}(t) + \mathcal{E}_{BUF}, \quad \forall t \tag{3.4}$$

This requirement should be tackled with some care. Despite the necessary condition expressed by Eq. (3.1) for the operation of a device, the instantaneous power supplied by the harvester $\mathcal{P}_{SCV}(t)$ is independent of the energy consumption rate $\mathcal{P}_{DEV}(t)$, and may follow completely different patterns and characteristic periods (as represented in Fig. 3.2, top). Often, the former is conditioned by the environmental conditions, while the latter is determined by the specific application requirements. Then, a properly sized energy buffer is required to absorb the temporary asymmetries between generation and demand. Basically, this temporal storage unit has to accumulate enough energy in favorable periods that it will be delivered in short bursts to the electronics when production is scarce.
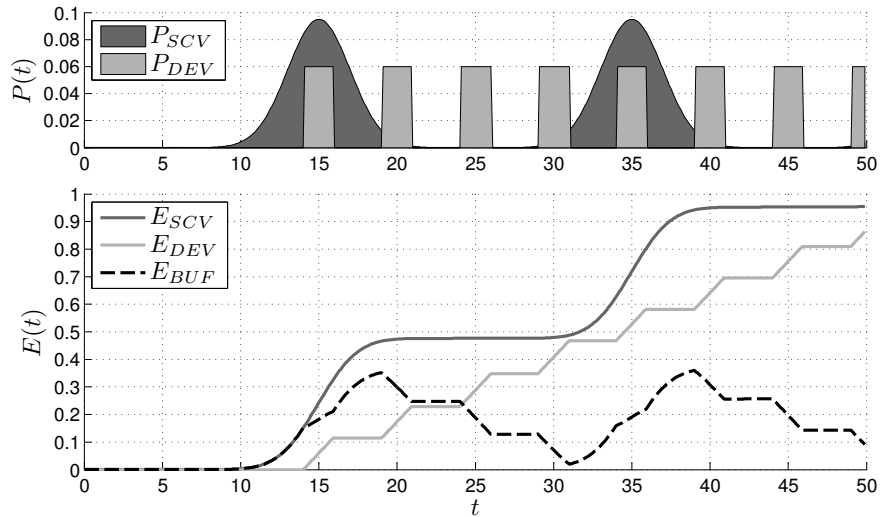
Figure 3.2. *Power production and demand, with accumulated energy associated.*

Provided that the condition of Eq. (3.4) is guaranteed, dimensioning energy generation and energy consumption can be addressed independently. Thus, Eq. (3.1) can be reduced to the simpler condition expressed by Eq. (3.5) in terms of average power, which must be satisfied within the characteristic operational cycles of the application. The problem of a wireless sensor network's sustain-

ability is then elucidated through a clear understanding of the energy generation patterns, the characterization of the operational energy demand and the proper dimensioning of the energy buffer.

$$\bar{\mathcal{P}}_{SCV} \geq \bar{\mathcal{P}}_{DEV} \tag{3.5}$$

3.1.2 Supply and Demand Balance

Dimensioning of the scavenger depends on the configuration and application requirements of the wireless sensor devices. Recalling the model described by Eq. (2.18), two different approaches can be followed to achieve Eq. (3.5)'s condition.

On the one hand, given a particular scavenger characterized by an average power $\bar{P}_{SCV}$, the network and application parameters can be tuned to meet Eq. (3.5). To illustrate this idea, in Fig. 3.3a the average current provided by the scavenger has been represented as a fixed horizontal plane. Eq. (3.5) is fulfilled for all bars that stay below the plane, that is, configurations for which device demand is lower than scavenger production. The black solid line delimits the region of parameter configurations that satisfy the above condition; this is called the *self-sustainable region*. The application settings may then be relaxed until the border line is crossed in any of the two directions indicated by the arrows, i.e., until the Eq. (3.5) condition is achieved. This is a common situation in which devices have hard mechanical constraints and therefore limited space available for accommodating the scavenger.

On the other hand, a fixed network and application configuration can be used to draw the upper bound of $\bar{P}_{DEV}$ and therefore select the right size for the scavenger. In this case, the bar representing the device's power is fixed in Fig. 3.3b, but the level of the scavenger plane is moved up until it reaches the required value above the bar level to satisfy Eq. (3.5).

3.1.3 Source Selection

The classification of energy harvesters can be based on the kind of energy they extract from the environment to produce electrical energy: mechanical, solar, thermal, electromagnetic, etc. Table 3.1 compiles the main off-the-shelf harvesting technologies with their power generation capabilities. When dealing with a new project, it would be good practice to first look-up similar tables to check the order of magnitude of power one can expect.

The availability of the energy source is without a doubt the major criteria for selecting the energy harvester. Other general properties to be considered include: physical features (power density, maximum voltage and current), mechanical characteristics (size, shape or weight), environmental protection, water resistance and temperature range, as well as other operational and maintenance requirements.

(a) *Fitting application parameters.*

(b) *(Re)dimensioning harvester.*
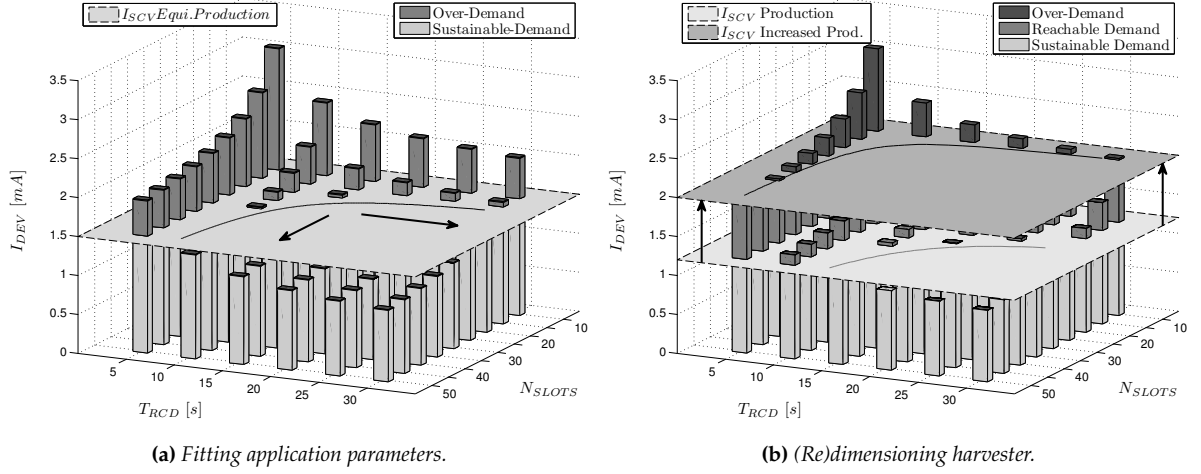
Figure 3.3. *Alternatives to balance device consumption and harvester production. Numerical values from Sec.2.5.3.*

Table 3.1. *Characteristics of various energy sources.* †*[79]* ‡*[53]* §*[38]*

Source	Conditions	Source Power	Harvested Power
Ambient Light†	Indoor	$0.1\,mW/cm^2$	$10\,\mu W/cm^2$
	Outdoor	$100\,mW/cm^2$	$10\,mW/cm^2$
Motion†	Human (0.5m - 1Hz)		$4\,\mu W/cm^2$
Vibration§	Industrial (120Hz - 0.1g)		$40\,\mu W/cm^3$
Thermal †‡	Human $\Delta T \sim 5°K$	$20\,mW/cm^2$	$30 - 60\,\mu W/cm^2$
	Industrial $\Delta T \geq 20°K$	$100\,mW/cm^2$	$1 - 10\,mW/cm^2$
RF †	GSM Base Station	$0.3\,\mu W/cm^2$	$0.1\,\mu W/cm^2$

3.1.4 Energy Buffer Selection

Choosing the capacity of the energy storage device and sizing the buffer can be accomplished by careful examination of the energy input and output patterns. When the energy income to the system is available only intermittently, the buffer size is determined by the amount of energy required to operate the device during scarce scavenging periods. In other applications, energy is more-or-less freely available at the system input, whereas the energy outcome occurs in spaced-out intervals. In both cases, the same basic considerations apply when dimensioning the storage device.

Once scavenger and device settings are balanced (following any of the two approaches in section 3.1.2), they may meet the criteria expressed by Eq. (3.5). Under intermittent conditions, the energy production above the average level during over-production periods $P_{SCV} > \bar{P}_{SCV}$ has to be buffered in order to power the device when production is scarce $P_{SCV} < \bar{P}_{SCV}$. This energy is delivered to the device on demand at the required rate $P_{DEV}(t)$.

As stated before, storage capacity is sized according to the maximum difference between energy production and demand curves (See Fig. 3.2). The estimated value may be contrasted with Tab. 3.2, to infer the physical dimensions required for storage.

Table 3.2. *Characteristics of various energy storage technologies. §[25] †[78] ‡[67]*

Source	Op.Voltage [V]	Density [mWh/cm^3]	Charge Cycles	Self.Discharge [%/month]
Lithium (primary) §	3.7	1300	–	<0.1
Li-Ion†	3 - 3.7	435	2000	0.1 - 1
Ni-MH‡	1.2	300	300	<30
Thin-Film†	3.7	50	1000	0.1 - 1
Super-Capacitor†	1.2 - 2.5	5	∞	100

Multiple technologies for storing electrical energy are available on the market: thin-film solid state batteries, capacitors, etc. The choice of the appropriate technology is highly dependent on the application. The following recommendations should be considered:

- As shown in Table 3.2, the energy density of batteries is much higher than that of supercapacitors. In situations where a supercapacitor may not be able to store enough energy, a battery is required. Sometimes supercapacitors can be an alternative to batteries. For example, when the ambient energy source is intermittent a supercapacitor may be used. Yet, special care should be taken with sizing, as the energy must be stored not just for peak power delivery, but to support the application for longer interruptions. Furthermore, the energy density is limited.

- Instantaneous power delivery is higher for supercapacitors, an interesting feature in applications with high current peak demands. Among them, the most representative are GSM devices. Other examples include sensors that require external excitement (e.g., vibrating wire sensors) and actuators (e.g., small electrical motors).

- Supercapacitors store energy by physical-charge storage, not chemically as in batteries; so, supercapacitors have an effectively infinite cycle life. This is a desirable feature for applications with many charge-discharge cycles. In turn, a battery's self-discharge is lower, which should be keep in mind when dealing with long-term cycles of inactivity.

- In some applications, supercapacitors are best used to support batteries, not to replace them. If the peak power needed exceeds the amount the battery can supply, e.g., for GSM calls, or for low-power transmission in cold temperatures, then the battery can charge the supercapacitor at a low rate and the supercapacitor can deliver the high power bursts. This arrangement also means the battery is never cycled deeply, extending battery life.

3.2   Related Work

A vast literature has emerged on Energy-Harvesting for Wireless Sensor Networks in the last few years. The promise of long-term, uninterrupted and self-sustained operation in a wide spectrum of applications has captured the interest not only of academia, but also the industry.

Energy-harvesting encompasses virtually most of the topics in the discipline of the electrical engineering, including scavenging components, transducers, rectifiers, converters, energy storage, etc. In addition, due to the commonly low amounts of harvested energy, autonomous networks require extremely low-power consuming communication and control protocols. Even the integration of such systems with sensing and identification components requires novel interconnect and packaging approaches. Energy harvesting is thus a wide-ranging, multi-disciplinary and challenging topic.

In what follows, some of the most active fields related to self-sustainable networks are briefly reviewed. The approach taken here is to refer the reader to well-established texts published on each subject. These texts can be used as a starting point for further study.

Energy Harvesting for Wireless Sensor Networks

To begin with, Vullers *et al.* [79] provide a comprehensive introduction to the topic. This work reviews different characteristics of several energy harvesting methods and discusses future challenges. More recently, Kim *et al.* [39] update the analysis of leading ambient energy-harvesting technologies (solar, thermal, wireless, and piezoelectric), discussing their applicability in the development of self-sustaining wireless platforms.

More focused on specific topics, Nasiri *et al.* [51] present a deep study on how photovoltaic (PV) cells in indoor scenarios can be dimensioned to fuel low-power electronic devices. Tan and Panda [69] reproduce an analogous approach by comparing scavenging techniques to capture the energy from thermal variations with PV-cells. Finally, D'hulst *et al.* [24] address piezoelectric vibration scavenging. This work concerns mainly the analysis and modeling of the vibration scavengers that can be used to obtain energy from kinetic sources such as motors, pumps, etc.

Buffering

Roundy *et al.* [63] review the main potential power sources for wireless sensor nodes. Well established sources such as batteries are reviewed along with emerging technologies. A comprehensive analysis of electrical energy storage technologies can be found in Chen *et al.* [18], and more recently in Boicea [13].

Regarding buffer management, Jiang *et al.* [36] present a multi-stage energy transfer system and discuss the relationships among system components, identifying hardware choices to meet applica-

tion requirements. The two-stage storage system consists of a supercapacitor (primary buffer) and a lithium rechargeable battery (secondary buffer). The energy transfer model is similar to the approach adopted in this work.

Network Management Policies.

Transmission and schedule network policies play a fundamental role in self-powered networks. A great research effort has been made to cope with different optimization problems. Assuming that nodes have some knowledge of the arrival processes that describe the harvested energy inflow or the input data, different cost functions are evaluated (maximizing throughput while keeping the system alive, maximizing PDR, minimizing retransmissions, etc.). A comprehensive survey about this topic can be found in Gunduz *et al.* [34], with further reading to be found in the included references.

Despite dealing with complex mathematical formulations, some of the assumptions made by most of these works in searching for tractability impose severe conditions on the applicability to real problems. In particular, adopted models presuppose that all data acquired must be transmitted [2], and this premise is valid only for very simple applications. In modern systems, most of the process is carried out locally, and the mapping from acquired data to transmitted information is not trivial.

System-level and Ad-hoc Approaches

The last family of works addresses self-sustainability from a more pragmatic point of view. The basic idea is to consider a particular application when trying to demonstrate the viability of the system for some suitable harvesting device.

Torah *et al.* [71] present a self-powered industrial application based on vibration harvesting. This work focuses on characterizing the amount of energy that can be obtained from an electric motor, but it does not outline how this energy is distributed among the application subsystems, and it therefore neglects to address how this energy spending can be optimized. Waterbury and Wright [82] deal with the same application and present a similar approach.

$$\cdots$$

As a general consideration, a system-wide view is lacking, as well as a much-needed discussion on whether the obtained energy is enough to power a device. This was in part the main motivation behind addressing this topic through a systematic, well defined energy consumption profile, as presented in Chapter 2.

---

[2]Referred to in Chapter 2 as *Simple Reporting Applications*

3.3   Networks Powered by Vibration Sources

This section makes use of the model described in Section 2.5 to demonstrate the feasibility of self-sustained energy devices in real-world industrial applications. With the aim of illustrating the methodology, the model is applied to a wireless sensor node featuring vibrational energy scavenging and low-power communications, such as those enabled by TSCH. Even though the section is presented as a show case, it is important to emphasize that it should be considered as a modeling guide, and the methodology developed here can be extended to a wide range of applications beyond this particular instance.

3.3.1   Predictive Maintenance of Rotary Machines

Rotary machines of all sizes are the propelling force behind many industries (refining, chemical, energy production, etc.). In the constant quest for better efficiency, increased machine speeds and reduced outages become necessary, which in turn put great emphasis on machinery health. Vibration monitoring as a part of preventive/predictive maintenance programs has proven to be highly cost effective. The analysis of motor vibrations has been widely used for early-detection of structural damage and for monitoring the health of the machinery [50]. The benefits include reduced production losses, enhanced efficiency, reliability, longevity of machinery and reduced maintenance costs, to name just a few.

Monitoring systems get periodic records from sensors installed in the machinery and they try to detect anomalies in the operation by means of spectral analysis. This type of analysis requires a large amount of raw data that would eventually collapse wireless communications. For this reason, it is desirable to compute and analyze the spectrum locally and transmit only those results of particular relevance. In fact, this is a classic example of an application in which all processing is moved to the sensor side, and events are only reported when a specific alarm is triggered. In this kind of system, some sort of periodic health monitoring is required, so as to guarantee that the device remains operative and carries out its tasks continuously. Taking into account the above considerations, this section presents a case study based on the following specifications:

- The vibration monitoring application is concerned mainly with the frequency content of the acceleration signals. As such, an $\mathcal{FFT}$ is required to find the dominant harmonics.

- A TSCH network deals with communications. This technology is particularly well suited for this application, as the same mechanism for managing the network is used as a keep-alive indicator: the fact that network synchronization is maintained implies the device is still operative.

- Finally, vibrational energy is harvested to power the device. Therefore, the same physical magnitude being monitored also provides the source of energy for operating this smart application.

3.3.2 Experimental Setup
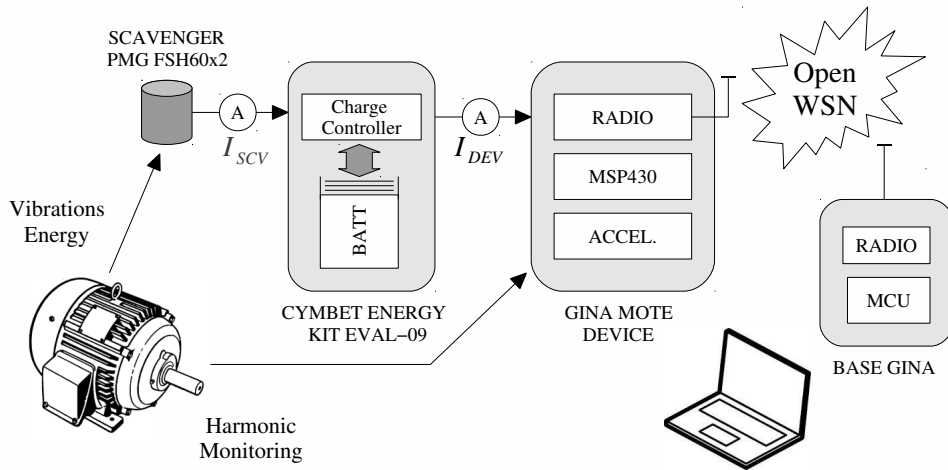
Figure 3.4. *Experimental setup for vibration energy sources.*

Fig. 3.4 shows the experimental setup designed to test this application. The core of the system is the GINA platform [49], featuring a Texas Instruments MSP430f2618 16-bit microcontroller, Atmel AT86RF231 IEEE802.15.4 radio, and the STMicroelectronics LIS344ALHTR 3-axis analog accelerometer. The selected vibration scavenger is the Perpetuum PMG FSH60x1. The device is attached to the case of the motor in a position that maximizes both the vibrations transferred to the sensor as well as those being scavenged. The energy storage unit is comprised of a lithium polymer battery and the EnerChip CBC915 Energy Processor. This battery controller unit is integrated into the Cymbet EVAL-09 Evaluation Kit. All experiments were measured with the NI9203 16-bit analog current acquisition module, with a resolution of $0.6\mu A$ per LSB.

3.3.3 Source Characterization

Mechanical vibrations of several industrial pumps were measured in order to characterize their amplitude and so the available energy. Several experiments were performed in 24-hour periods to evaluate the variability over time.

Figs. (3.5a) and (3.5b) show a 1-hour chunk of the spectrograms obtained. In this specific example, the maximum peak of the spectrum was found at 60Hz. At this frequency, the measured acceleration amplitude was almost constant in time and was found to have a value of $\overline{a} \pm \sigma_a = 0.098 \pm 0.002 \, [g]$.

The position of the main harmonic is particularly important, because the selected harvester should have its resonator frequency matched with the source. As shown in the pictures, electrical motors may have other harmonics that are often generated by the power line itself. The information con-

(a) *Spectrogram of vibrations during one hour of monitoring.*

(b) *Spectrogram colormap projection.*

(c) *Amplitude mapped to the current supplied by the harvester.*
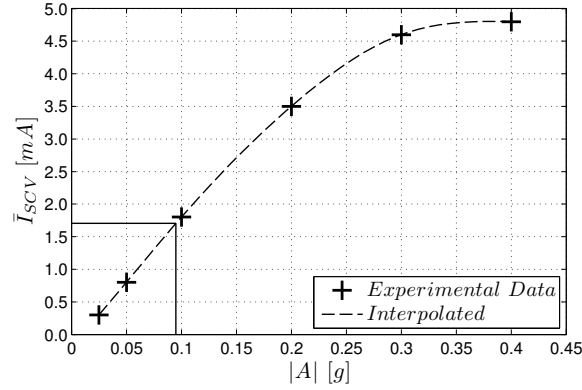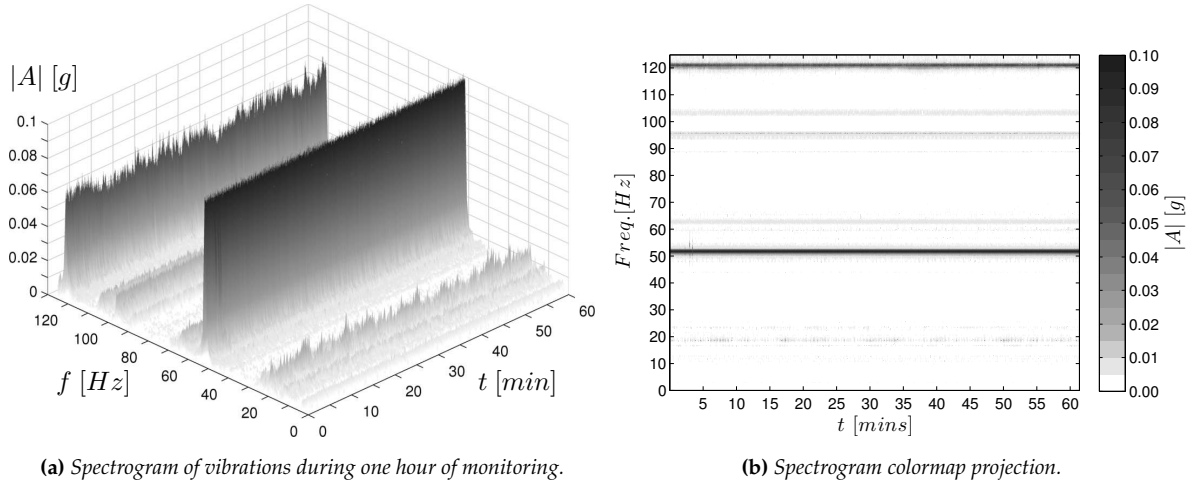
Figure 3.5. *Vibrational energy source characterization.*

tained in these secondary harmonics is essential for health analysis, but for harvesting purposes, the energy contained in the most prominent peak is obviously of greater interest.

As mentioned before, the Perpetuum PMG FSH60x1 harvester was used for the experiments. Fig. 3.5c shows the mapping between measured acceleration and the current produced by this device. As highlighted, the output remains above $1.5\,mA$ when attached to a vibration source of $0.1\,m/s^2$.

3.3.4 Application Settings and Adaptive Operation

The current budget obtained in ideal conditions for this application is $\bar{I}_{SCV} \approx 1.5\,mA$ (See Fig. 3.5c). This value is taken as the nominal system reference. On the basis on this expected outcome, Figs. 3.6a and 3.6b show the suitable subset of parameters for making the application self-sustainable [3].

The information contained in both figures is the same, but with different views. In Fig. 3.6a,

---

[3]This example follows the second approach defined in Sec. 3.1.2, as the application settings are configured to meet the production expected from a given scavenger.

(a) *3D-Bar representation.*

(b) *Selection chart. Arrow marks self-sustainable region.*
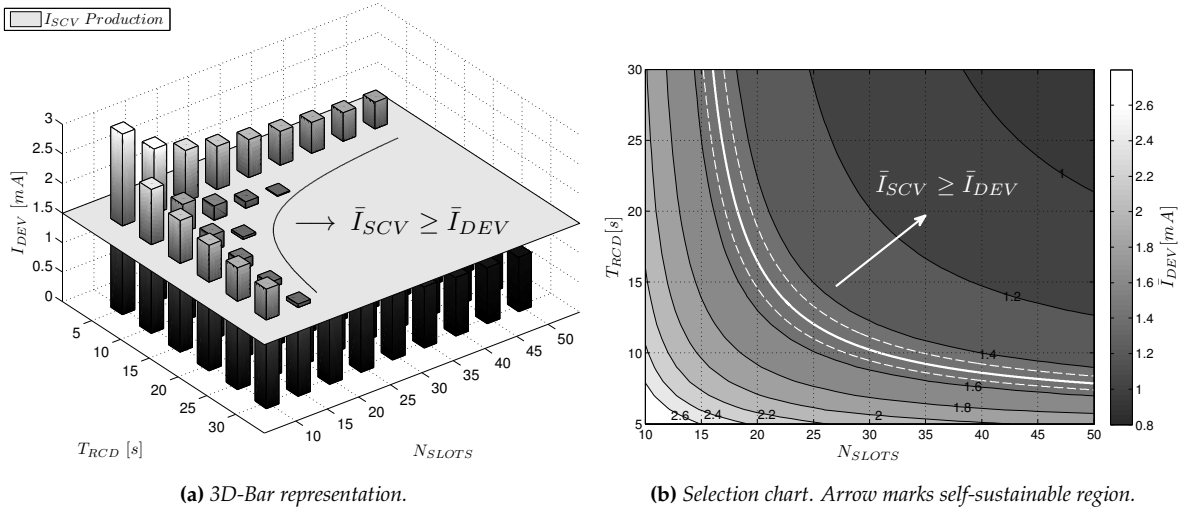
Figure 3.6. *Selection of self-sustainable application settings.*

bars represent the device demand for different settings, and the harvested current is depicted as a flat, equipotential surface. Bars below the plane belong to the self-sustainable region. For strictly illustrative purposes, the limit of this region has been projected on the harvester surface (black line).

In Fig. 3.6b, bars have been replaced by a colormap to improve readability. In this case, the self-sustainable region is delimited with a white curve. Given the estimated production of the specific harvester, the white line separates suitable combinations of parameters that guarantee autonomous operation in standard conditions.

Harvester Efficiency

In some applications, the incoming energy to the system is only available intermittently (e.g., pumps working on-demand). In other cases, the scavenger cannot provide the expected energy because something changes in the environment (e.g., small modulations in the harmonic frequency). In both cases, the application should be able to dynamically adapt to the new conditions so as to remain energetically sustainable. This can be accomplished by selecting a suitable control parameter and mapping the consumption to the expected energy income.

For vibration scavengers attached to rotary machines, the main cause of lost efficiency is the mismatch with the vibration frequency. These kinds of devices have a narrow band response, and any deviation from the resonance frequency reduces the outcome significantly. Other sources of variability are the quality of the attachment and aging. Despite this, the magnitude of vibrations is constantly monitored and application settings can conform to new conditions, thus recovering viability.

(a) *Time scheduling under variable harvested current.*   (b) *Adaptive harmonic analysis for duty-cycle motor operation.*
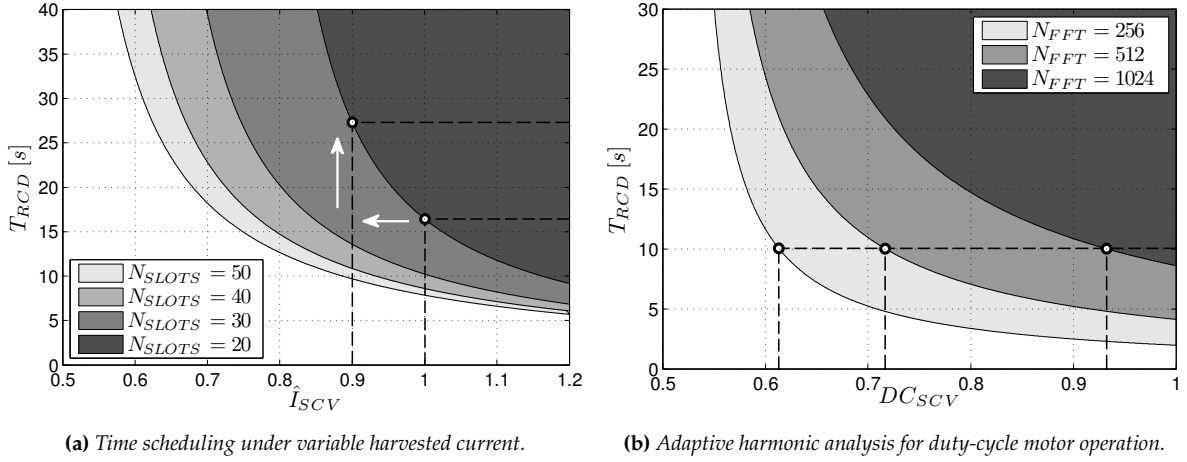
Figure 3.7. *Power consumption control through dynamic application settings.*

Following with the TSCH example, the number of slots is fixed once the network has been established (the length of the slotframe cannot be changed without involving a cascade of changes in all the nodes). *Active* slots could be deactivated and converted to *Sleep* slots, therefore reducing the throughput of the node. However, this incurs a considerable energetic cost to the network, as it might trigger rescheduling in other nodes.

A possibly more suitable parameter for regulating power consumption is the time lapse between records T_{RCD}. In order to dynamically adjust energy consumption to variations of the scavenged energy, the application needs to keep track of the amplitude of the fundamental vibration harmonics and estimate the expected input current for the next cycle accordingly.

Fig. 3.7a shows the operational regions for different network settings. To select the interleaving time, the system maps the analyzed vibration amplitude to get the expected normalized current (from Fig. 3.5c). This value is used to determine the timeout for the next wake-up by finding the T_{RCD} in the limit of the region. In the figure, gray areas represent the feasible zones for some selected network configurations. The system must react to a reduction in the harvested current (horizontal arrow) by increasing the time elapsed between records until reaching the self-sustainable region (vertical arrow).

Summarizing, T_{RCD} provides a fine control mechanism for adjusting average power consumption to variations in the scavenged energy. In this case, the penalty for the application's sustainability is that more time elapses between successive records and, thus, it compromises the performance of the monitoring application.

Motor Duty-cycling

When rotary machines do not operate continuously, the average power obtained is necessarily lower. Nevertheless, when the motor is halted, continuously monitoring the health through vibrations does not make any sense at all. The problem that arises is the question of when the sensor device should resume monitoring.

The $\mathcal{FFT}$-analysis provides an easy way to obtain the instantaneous motor status. When the machine is switched off, the magnitude of the fundamental harmonic experiences a significant diminution, and the device can reduce its activity until the machine is restarted. In this case, the number of points for computing the $\mathcal{FFT}$ can be reduced while keeping the interval monitoring time fixed. Obviously, this would compromise the quality of the harmonic analysis in normal operation. But, in the idle state, the only reason to maintain harmonic tracking is to restore normal operation when the motor wakes-up again, and for this purpose the quality factor of the harmonic is secondary.

Fig. 3.7b shows how a reduction in the number of points of the $\mathcal{FFT}$ allows the device to remain in a self-sustainable region, even with a significant reduction in the current scavenged. This is an effective solution that does not compromise the application latency.

3.3.5 Buffer Selection

Duty-cycled operation is managed by temporally buffering overproduction during the active periods of the energy source (the harvester must be properly sized). The size of the buffering device depends fundamentally on three key aspects. First, applications settings, which in the end determine the device's demand; second, the required autonomous operation, i.e., the longer supply interruption that can be absorbed; and third, the particular technology chosen (see Table 3.2 for the expected energy density). Table. 3.3 offers three commercial alternatives for the specific application settings of Fig. 3.7b. The table contains the expected operational life in the absence of energy production, provided that the buffer is initially fully recharged, together with the physical dimensions of the device. In this case, devices have been selected with a small form-factor that is compatible with many standard embedded devices, but the particular choice may depend on the mechanical constraints.

Table 3.3. *Unattended operation limit for different technologies.*

Technology	Type [Ex.]	Capacity [mWh]	Dimensions [mm]			Running Time [h]		
			L	D	H	N_{FFT}=256	N_{FFT}=512	N_{FFT}=1024
Li-Ion	CR2032 [†]	210		20	3.2	228.3	195.3	150.1
Super-Cap.	Axial 25F [§]	25		16	26	27.2	23.3	17.9
Thin-Film	1mAh [‡]	4	25		0.2	4.3	3.7	2.9

† Maxel CLB2032 3V-70mAh Rechargeable Coin-Cell.

‡ Inf.Pow.Sol. Thinergy MEC201-10 Rechargeable Solid-State Cell (Flexible).

§ Maxwell HC-Series BCAP0025 Supercapacitor.

3.4 Networks Powered by Solar Energy

This section continues with the model described in section 2.5, using it as a basis for building a second, real-world application. In this case, a wireless sensor node is used to monitor in real time the health of a power line transmission tower. This application uses solar energy as a source for the device, taking advantage of the open-air exposure of the towers. This use case serves as an example to illustrate the alternative approach defined in Sec. 3.1.2, as now the harvester will be sized accordingly to the system requirements.

3.4.1 Monitoring Power Line Towers

Tower failures are problematic. They produce unplanned downtimes and energy delivery losses, which may affect businesses and citizens. The main reasons for transmission tower failure can be attributed to structural damage due to corrosion, mechanical damage due to impacts and strong winds, or structural overloading due to ice and snow on the conductor lines. Moreover, minor local defects can produce structural damage (for example, when the small traversal bars are destroyed or some other structural elements of the tower are subject to bending). Such damage can also cause a sudden tower collapse. If the structure begins to deteriorate, corrosion advances exponentially, and in a few years it can oxidize the tower to the point of failure. Furthermore, if the tower deterioration advances, the repair time, labor and material cost for repairing the tower increase significantly [62]. But these effects can be mitigated through preventive maintenance at the right time.

The periodic inspection of transmission towers is hence necessary for ensuring the reliability of electric service to customers. However, due to the spread and large number of towers, and due to towers often being located in places with difficult access, it is difficult to assess the structural integrity of the network from the ground. To make a qualitative assessment, it is usually necessary to conduct aerial inspections, through which experts grade tower conditions. This method is however very subjective, and evaluating the remaining strength and service life of towers is still difficult and prohibitively expensive.

Subsequently, ground inspections are only performed on a fixed schedule in very specific areas. A field engineer drives to a particular tower on a specific date and inspects every element of the tower. Servicing companies and utilities will inevitably spend valuable time inspecting towers that would otherwise keep functioning for a long time.

In the present scenario, it is desirable to equip each tower with multiple sensing elements which provide information about the tower's structural state of health in real time. The data is transmitted to a central system, where is evaluated, analyzed and stored for predictive maintenance. The potentially difficult access to the towers requires wireless communications as well as very-low maintenance. But, above all, the need to replace batteries must be avoided.

3.4.2   Experimental Setup
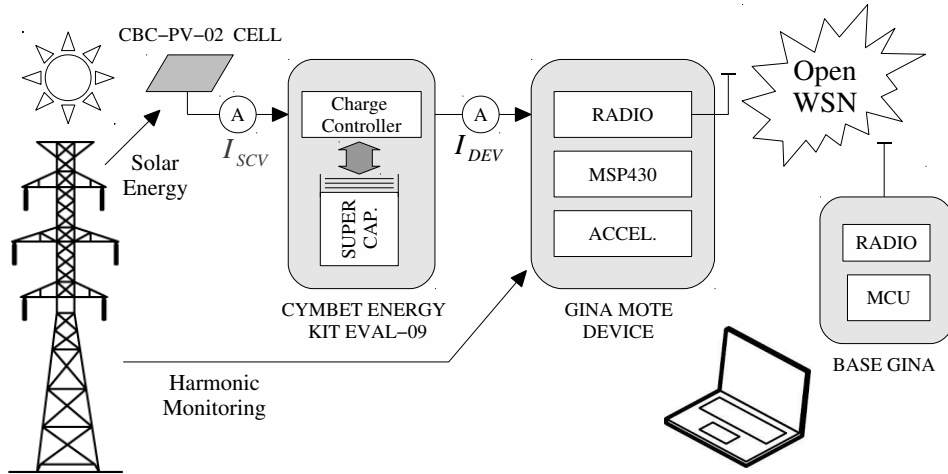
Figure 3.8. *Experimental setup for solar energy sources.*

Fig. 3.8 shows the experimental setup designed for this application. This setup is based on the same platform as Section 3.3, namely a GINA [49] device connected to a TSCH network. In this case, the vibration scavenger used in Fig. 3.4 has been replaced by a small photovoltaic cell. The energy storage unit has been improved by a Maxwell Supercapacitor, used to manage continuous charge and discharge cycles that result from the solar energy intermittence. The CBC915 charge controller allows for simultaneous management of lithium batteries and supercapacitors, which simplifies the tests procedures. This controller unit also allows for switching between different energy-harvesting sources.

3.4.3   Analyzing the Energy Source

It is a well known fact that solar energy is available intermittently. More concretely, solar irradiation is characterized by a double periodicity: daily cycles (day-night) and seasonal cycles (winter-summer). This double periodicity can be easily appreciated in Fig. 3.9, where the monthly evolution of solar irradiation in Los Angeles (California) is presented for the 2001-2010 period. This data is publicly available, and similar information can be obtained for other locations [3].

Aside from the monthly evolution, the same figure indicates the day-night variations, which are represented in the vertical direction for each specific month and split into slots of 1 hour. To make this evolution understandable, data is projected in two orthogonal directions. In the first one, daily irradiation has been aggregated (the figure below the scale-colormap). This plot shows the monthly evolution, i.e., the seasonal variations, averaged per day, for the expected energy income.
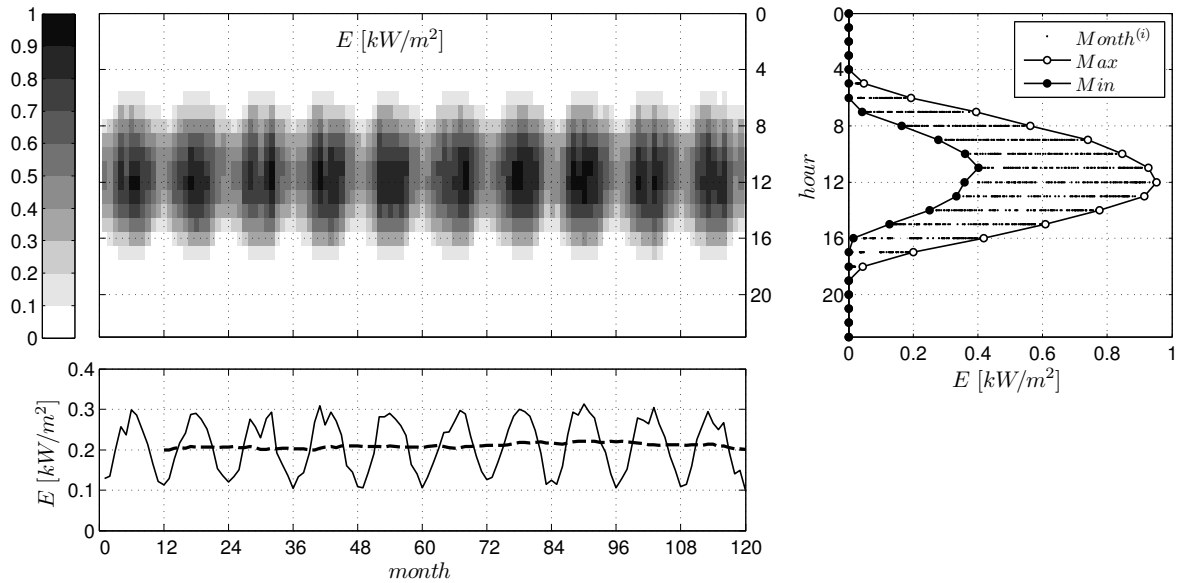
Figure 3.9. *Solar irradiation in L.A.City over the period 2001-2010 from different views. Top-Left: monthly irradiation colormap. Each point is the average irradiation at the same hour (vertical dimension) over the specific month (horizontal). Top-Right: projection of all irradiation values, showing the variability in the full period. Bottom-Left: daily average projection. Each value is the monthly average irradiation (dashed line shows year average).*

The second one is a projection of the hourly power of every single month in the period (right side). This figure captures with a single snapshot the historic daily behavior of solar irradiation. Obviously, some months have longer days than others, and even the same month can see irradiation change from one day to another; but the interruption of energy production at night is inevitable.

In light of these patterns, this proposal handles the double periodicity in two different ways: seasonal variations are absorbed by dynamically adapting the application to the expected energy income, while daily energy over-production is stored in a super-capacitor for powering the system at night. The next two sections lay out the procedure in detail.

3.4.4   Harvester Sizing and Adaptive Operation

When the available energy is highly variable (as solar irradiation is) the selection of a suitable combination of the harvester size and application settings is complex, as the system should be designed to stay alive even in worst case conditions. This becomes especially relevant when the application needs to dynamically adapt to new, non-predicted conditions, while still remaining energetically self-sustainable.

Looking at the historic evolution of the averaged irradiation (Fig. 3.9, bottom), the minimum expected value for the daily average power is $0.1kW/m^2$, corresponding to the worst winter period

on the record [4]. Based on this observation, and considering the efficiency of the selected solar panel technology (typically $\eta_{Panel} \approx 15\%$) and the DC-DC converter ($\eta_{dc} \approx 50 - 80\%$), it can be expected that the power production per area unit of the system can be estimated in $\hat{P} \approx 0.8\ mW/cm^2$. This value, once converted according to the Section 2.4.1 rules (with the losses due to DC/DC conversion accounted for in the efficiency factor η_{dc}), corresponds to $\hat{I}_{SCV}=0.32\ mA/cm^2$ at 2.5V [5].

With this value in mind, the next step is to understand the current requirements of the application. Fig. 3.10 shows the operational regions for different network configurations (different schedules that can provide different levels of QoS or bandwidth). Again, the number of active slots is fixed once the network has been established and, therefore, the time between consecutive measurements T_{RCD} would serve as a more appropriate control parameter in accordance with energy availability.

At this point, a key aspect to consider is the asymptotic behavior of Eq. (2.18). Even when using the very high values of T_{RCD}, the offset term due to network maintenance and system background functionality makes it almost impossible to obtain further energy savings. This is the asymptotic limit below $0.5mA$, which is noticeable in Fig. 3.10. It is important to understand that this value is a system limit fixed by the technology used. Therefore, it determines the absolute minimum size of the solar panel, meaning that the harvested current must be above this value in worst case conditions. This is a good example of the necessity for a system-wide view in the wireless device's design flow: a power condition is imposed by radio technology and emerges as a mechanical system constraint through the size of the solar panel.

Bearing in mind the minimum value needed to operate the network, and recalling that expected current per unit are is $\hat{I}_{SCV} \approx 0.3\ mA/cm^2$, a small form-factor panel of $2cm^2$ guarantees a current of $I_{SCV} = 0.6mA$ in worst case irradiation periods. Beyond this value, the system can keep track of the variations of the energy scavenged and estimate the current income for the next cycle, in order to operate in a more aggressive power mode according to the season.

In Fig. 3.10, the gray areas represent the feasible zones that satisfy the condition $I_{DEV} \leq I_{SCV}$. The device is configured in a low-power mode during winter. In this mode, the device analyzes the state of the tower every 30 seconds, which satisfies the sustainability condition for a current of $I_{SCV} \geq 0.6mA$ (in the direction of top-left arrow). In summer periods, when the expected incoming energy is around 3 times higher, the device can be switched to high performance mode. In this mode, the device takes a record every 5 seconds, which is also sustainable for currents above $I_{SCV} \geq 1.8mA$ (bottom-right arrow).
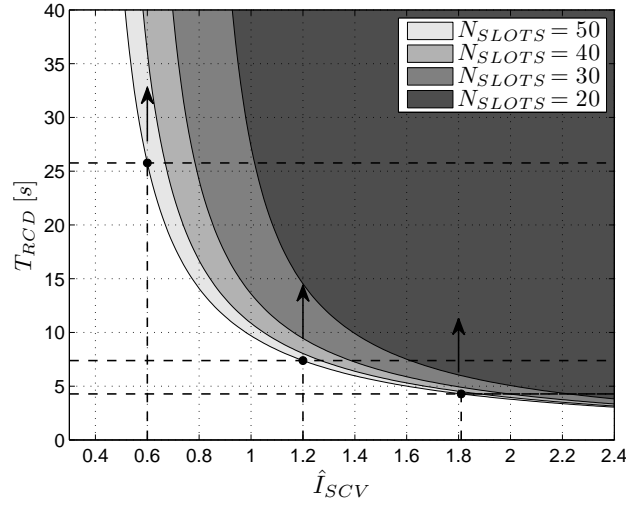
Figure 3.10. *Dependence of the time between records and the harvester current for different network configurations. Gray areas represent the feasible zones that satisfy the condition $I_{DEV} \leq I_{SCV}$. The device is configured in a low-power mode during winter. In this mode, the device analyzes the state of the tower every 30 seconds, which satisfies the sustainability condition for a current of $I_{SCV} \geq 0.6mA$ (in the direction of top-left arrow). In summer periods, when the expected incoming energy is around 3 times higher, the device can be switched to high performance mode. In this mode, the device takes a record every 5 seconds, which is also sustainable for currents above $I_{SCV} \geq 1.8mA$ (bottom-right arrow).*

3.4.5 Energy Buffer Sizing

Night intermittence is managed by temporally buffering to a *super-capacitor* the over-production during the central hours of the day. Super-capacitors support a virtually unlimited number of charge and discharge cycles, therefore making them particularly suitable for this application (See Table 3.2 and discussion at the end of Section 3.1.4) One of the drawbacks of super-capacitors is their relatively high self-discharge ratio, but daily replenishment make this effect negligible. In addition, super-capacitor technologies offer a good trade-off between energy density and peak current, the latter being desirable for radio transmissions.

For buffering purposes, worst case scenarios occur when the devices operate in high performance mode, as a higher amount of energy needs to be stored in order to power the system at night. So, in contrast to the sizing method, it is important in this case to consider the periods when the maximum amount of harvested energy is expected.

Fig. 3.11 shows the historic irradiation from Fig. 3.9 mapped onto the power generated by a $2cm^2$ cell (obtained after the DC/DC regulator). In the high performance settings, the average demand

---

[4]This value is roughly 10% of the power received with maximum irradiation (See Tab. 3.1)

[5]Can be compared with the $1.5\ mA$ reference value of the vibration harvester at $0.1\ g$.
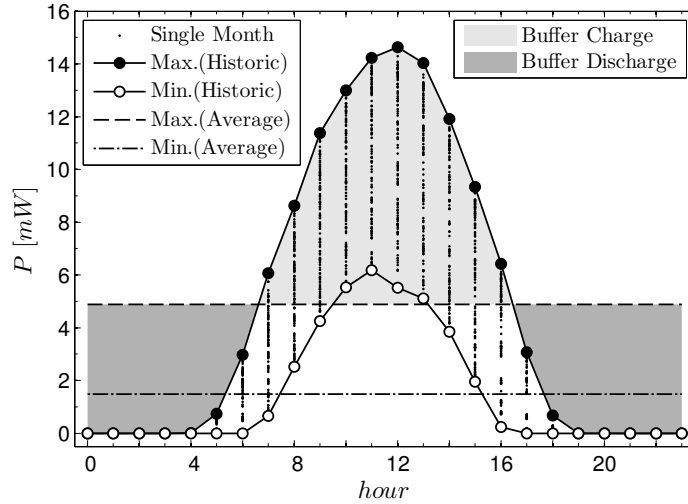
Figure 3.11. *Monthly-Daily-Hourly variations over the 2001-2010 period. Daytime over-production (light gray area) must be stored to cover device demand during the under-production periods (indicated by the dark gray area).*

from the device is close to $4.5\ mW$. Daytime over-production is depicted with the light gray area inside the maximum production (black-solid dots). The device demand (dashed-line) must be stored for use during the under-production periods, indicated by the dark gray area. In this example, the amount of energy to be buffered is $E_{Buff} \approx 58\ mWh$, which requires a $75\ F$ super-capacitor [55]. Specifically, we use the Maxwell BCAP0100 with a capacity of 100F in $17cm^3$ (L=45mm d=22mm), used at 75% of its capacity [6].

This result demonstrates how solar harvesting and super-capacitor technologies can be combined into a reduced form factor that enables reasonably small perpetual devices.

---

[6]From Tab. 3.1, an estimated value of at least $12\ cm^3$ is required for $58\ mWh$.

Discussion

Using the thread of sustainability as the main driver, this chapter makes use of the analytical model developed in Chapter 2 as a tool for facilitating the application configuration and scavenger selection at pre-deployment stages. The proposed approach is oriented toward providing concise answers to the main concerns about energy harvesting. The resulting methodology can be outlined in six simple steps: i) Find a source of energy in your environment. ii) Measure the magnitude of the available energy, characterize its cycles and analyze possible patterns. iii) Run a simulation within the feasible limits of application parameters. iv) Select a suitable harvester that is sized according to both the energy available and the first estimate of the application's consumption. v) Fine-tune the application parameters according to the selected harvester. vi) Select an appropriate buffer.

With the aim of demonstrating its practical usefulness, the presented methodology has been applied to two real application scenarios. Both case-studies looked at a given industrial wireless network application, and considered their radio technology, as well as sampling and processing requirements, all of which were needed to provide an accurate estimate of energy demands. This information was then used to determine what scavenger and super-capacitor was required to make it self-sustainable. Furthermore, the parametrized model was used to properly configure the application in order to enable different modes of operation in case of varying conditions.

Additionally, this chapter has prompted an evaluation of the main off-the-shelf technologies related to: energy-harvesting as energy sources (vibration, thermal and solar harvesters); energy storage (batteries, thin-film batteries and supercapacitors); and energy conditioning (ultra-low power profile DC/DC converters, regulators and inverters).

In summary, by remaining always within the framework defined in Section2.1, we have solved the sustainability problem of a wireless sensor on the basis of having a clear understanding of the energy production patterns, the characterization of the energy demand throughout the device's operation, and the proper dimensioning of the energy buffer. This approach ensures the right balance of the system and enables the potential cost optimization of product development.

More interestingly, this chapter uses these two real-word applications to launch an exploration of the quality of service concept that is applied to a monitoring application. Specifically, this chapter responds to a recurrent demand by system-developers: continued performance when energy is scarce. In this case, the performance (interpreted as quality of service) is measured in terms of how often the system carries out monitoring operations, whereas the constraint is self-sustainability.

Chapter 4

Lossy Compression Methods

The rationale behind data compression is that one can obtain compression in exchange for some reduction in the energy spent on transmission. The common thinking among researchers and implementers is that compression is always a good choice, because the major source of energy consumption in a sensor node comes from the transmission of the data. This is why temporal compression has been widely investigated in the field of wireless sensor networks, where energy efficiency is a crucial consideration due to the constrained nature of the sensor devices.

Lossy compression is deemed a viable solution, as the imperfect reconstruction of the signal is often acceptable. As we shall see in this chapter, this may allow some important savings. Even more, some compression algorithms inherently entail filter capabilities, and discarding some (noisy) information can even be desirable.

Most importantly, with some lossy algorithms it is possible to adjust compression to some (application dependent) maximum error tolerance, allowing additional gains in terms of compression, but sacrificing some quality in the reconstruction. This trade-off is seldom quantitatively exploited.

This chapter tries to fill this gap first by evaluating a number of lossy compression methods selected from the literature, and second by analyzing their performance in terms of compression efficiency, computational complexity and energy consumption. With this aim, the overall energy consumption and signal representation accuracy is discussed as a function of the most relevant signal statistics.

4.1   Introduction: Compression for Energy Constrained Devices

$\mathbf{G}$IVEN the high amount of energy spent on communications, it is natural to try to reduce radio energy. Compression seems a self-evident choice, as it directly reduces the size of data to be communicated. These assumptions have been widely accepted by the community. Only a few works in the literature look into the adverse impact of compression, namely the cost of compression itself. Current processor technologies seem inexpensive in terms of energy, but when the algorithms complexity increases, their contribution should be taken into account. This fact suggests (and the rest of this chapter aims to demonstrate) that compression should be addressed with some care. Indeed, a systematic analysis is required in order to evaluate the real impact of compression on overall energy consumption.

This chapter focuses on the energy saving opportunities offered by *lossy temporal compression* of data. With lossy techniques, the original data is compressed by discarding some of the original information in it. Hence, at the receiver side, the decompressor can reconstruct the original data only up to a certain level of accuracy. Depending on the application, this small inaccuracy in the reconstructed signal may be acceptable.

In terms of compression ratio, lossy compression makes it possible to trade some reconstruction accuracy for some additional gains. Note that these gains correspond to further savings in terms of transmission needs. Thus, lossy compression provides an additional flexibility, as one can tune the compression ratio based on energy consumption criteria, even though this means the loss of some fidelity in the reconstruction.

In the scenario of wireless sensors, compression ratio is still a fundamental performance metric for compression algorithms. However, the cost function is determined mainly by the energy consumption, and the accuracy of the recovered signal can be interpreted as a constraint. This relationships create a particular perspective, which will be detailed in the last section of this chapter.

4.1.1   Metrics for Lossy Compression

As a basis for the analysis, as well as to provide a more formal treatment of all this concepts, the following definitions will be considered in the remainder of this chapter:

$\mathscr{D}$. **Compression Ratio η:** Given a finite time series $x(n) \doteq x_1, \cdots, x_n$ and its compressed version $\hat{x}(n)$, η is the ratio between the number of bits used to represent the compressed time series $N_b(\hat{x})$ and the number of bits of the original one, $N_b(x)$. With this definition, compression is achieved if $\eta < 1$.

$$\eta = \frac{N_b(\hat{x})}{N_b(x)}$$

$\mathscr{D}$. **Reconstruction error $\varepsilon(\mathbf{n})$:** Given a discrete time series $x(n)$ and its compressed version $\hat{x}(n)$, the reconstruction error at time $n \geq 1$ is defined as the deviation between the original sample and the recovered one, measured by the Euclidean distance metrics.

$$\varepsilon(n) = |x(n) - \hat{x}(n)|$$

$\mathscr{D}$. **Error Tolerance $\varepsilon_{\mathbf{T}}$:** The maximum permitted error at the receiver end for all terms in the series. It must satisfy the following condition for all values of n:

$$\varepsilon_T \geq \varepsilon(n), \ \forall n$$

4.1.2 Processing vs. Transmission Energy Tradeoffs

Standard compression algorithms are aimed at reducing storage size, not saving energy. As a result, compression ratio is their fundamental metric. This work focuses instead on energy savings as the primary metric. Thus, the relation between compression and energy must be clearly stated. To this end, it is important to define some useful concepts.

$\mathscr{D}$. **Compression Energy $E_{\mathbf{C}}$:** The energy used to accomplish the compression task. This contribution accounts for only the CPU operations associated with compression algorithms, without considering the additional costs related to other tasks of the micro-controller. Specifically, $E_C(x)$ is the processor energy for compressing a data series.

$\mathscr{D}$. **Transmission Energy $E_{\mathbf{T}}$:** The energy consumed for transmission, accounting for the radio chip characteristics and the protocol overhead. $E_T(x)$ is the total energy for transmitting the original series, $x(n)$, while $E_T(\hat{x})$ is the energy for transmitting the compressed version, $\hat{x}(n)$.

$\mathscr{D}$. **Total Energy E:** The sum of the energy consumption for compression $E_C(x)$ and transmission of the compressed data $E_T(\hat{x})$. It is equal to $E_T(x)$ in the uncompressed case.

$$E(x, \hat{x}) = E_C(x) + E_T(\hat{x})$$

$\mathscr{D}$. **Compression Energy Balance $E_{\mathbf{B}}$:** The difference between the cost $E_T(x)$ associated with transmitting $x(n)$ uncompressed and the combined cost $E(\hat{x})$ of processing and transmitting $\hat{x}(n)$ compressed. This can be expressed as the difference between the savings in transmission and the loss in compression.

$$E_B(x, \hat{x}) = \Delta E = E_T(x) - E(x, \hat{x}) = (E_T(x) - E_T(\hat{x})) - E_C(x)$$

$\mathscr{D}$. **Compression Energy Gain $E_{\mathbf{G}}$:** The ratio between the energy spent on transmission with no compression, $E_T(x)$, and the total energy spent on compression and transmission, $E_{CT}(\hat{x})$.

$$E_G(x, \hat{x}) = E_T(x)/E_{CT}(x, \hat{x})$$

A compression method is said to be *efficient* when its energy balance is positive; that is, the overall cost for compression and transmission of the compressed data is strictly smaller than the cost associated with transmitting data uncompressed. Formally, the *efficiency condition* is expressed as:

$$E_C(x) + E_T(\hat{x}) < E_T(x) \tag{4.1}$$

Taking the inequality on both sides of equation Eq. (4.1), dividing it by $E_T(x)$, and then rearranging the terms leads to:

$$\frac{E_T(x)}{E_C(x)} > \frac{1}{1 - E_T(\hat{x})/E_T(x)},$$

Assuming that the transmission energy is proportional to the number of bits to be transmitted, and that it is parametrized by the transmission energy per bit E_T^b, then $E_T(x) \cong E_T^b \cdot N_b(x)$ and $E_T(\hat{x}) \cong E_T^b \cdot N_b(\hat{x})$. By substituting on the right-hand side of the equation and using the definition of the compression ratio η, the efficiency condition of Eq. (4.1) can be expressed in terms of η:

$$\frac{E_T(x)}{E_C(x)} > \frac{1}{1 - \eta},$$

The energy for compression is decomposed into the product of three terms: E_I, the energy spent by the micro-controller per instruction; N_I^b, the number of instructions performed by the compression algorithm per (uncompressed) bit of $x(n)$; and $N_b(x)$, the number of bits composing the input signal $x(n)$. Using once again $E_T(x) \cong E_T^b \cdot N_b(x)$, Eq. (4.1.2) can be rewritten as:

$$\frac{E_T^b N_b(x)}{E_I N_I^b N_b(x)} > \frac{1}{1 - \eta},$$

The above inequality can be rearranged so that the quantities that depend on the selected hardware architecture appear on the left-hand side, leaving those quantities that depend on algorithmic aspects to remain on the right-hand side, resulting in Eq. (4.2):

$$\frac{E_T^b}{E_I} > \frac{N_I^b}{1 - \eta} \tag{4.2}$$

The right-hand term of Eq. (4.2) explicitly shows two key parameters in order to characterize a compression algorithm with an energy optimization objective. As expected, the compression ratio η takes part of this metric. Moreover, the computational complexity of the algorithm also takes part on the compression efficiency through N_I^b, a parameter representing the number of instructions required to process one bit.

In turn, the left-hand side of Eq. (4.2) captures the balance between radio and processor technologies for the specific platform. In current low-power wireless devices, the energy cost of transmitting a single bit is approximately the same as that needed for processing a few hundred operations [58] [5]. This is an important value to keep in mind, as it provides an effective means for measuring the allowable complexity of an energetically efficient compression algorithm.

4.1.3 Generation of Test-Bench Signals

Compression performance depends not only on the selected algorithm, but also on the specific signal. To fairly compare different algorithms, they must be evaluated over different types of signals and multiple realizations of each type.

Characterization of Signals

The first matter to address in a systematic study of compression algorithms is the classification of the input signals. As a general rule, the more aleatory a signal is, the less compressible it is. In lossy compression, the achievable compression ratio is adjustable, to some extent. Yet, in lossy compression, the penalty to be paid for randomness is reconstruction accuracy. Therefore, randomness is a parameter that must be taken into account for signal classification.

Furthermore, temporal correlation may help to reduce the information required for recovering the original data series. Intuitively, (temporal) similarity is evidence of redundancy. This means that a suitable technique may be used to remove some information without losing significance. As detailed in the next section, many compression algorithms take advantage of this fact directly or indirectly. Not surprisingly, correlation and randomness are closely related. This relationship suggests that correlation can be a good candidate for signal characterization when dealing with compression algorithms.

This chapter follows this approach. For this reason, it is very convenient at this point to establish the following definitions:

$\mathscr{D}$. **Autocorrelation** ρ: The cross-correlation of a signal with itself. Given a stationary discrete time series $x(n)$ with $n = 1, 2, \ldots, N$, the autocorrelation is defined as:

$$\rho_x(n) = \frac{\mathrm{E}\left[(x(m) - \mu_x)(x(m+n) - \mu_x)\right]}{\sigma_x^2}$$

where μ_x and σ_x^2 are the mean and the variance of $x(n)$, respectively.

$\mathscr{D}$. **Correlation Length** λ: For a given discrete time series $x(n)$, the correlation length is the smallest value such that the autocorrelation function of $x(n)$ is smaller than a predetermined threshold ρ_{th}. Formally:

$$\lambda = \underset{n>0}{\mathrm{argmin}} \left\{\rho_x(n) < \rho_{\mathrm{th}}\right\} \tag{4.3}$$

The novel approach here, which had not previously been tackled by the literature, is to introduce signal correlations when evaluating compression algorithms and, more specifically, to identify correlation length as a key parameter. The remainder of this chapter aims to demonstrate the suitability of this approach.

Generation of Synthetic Stationary Signals

A fair comparison of algorithms requires a test-signal database with multiple realizations for each target correlation (assuming correlation to be a characteristic parameter of each type of signal). The number of realizations must be large enough to guarantee statistical significance, and therefore some systematic approach to build a database is desirable. The approach adopted here is to generate a set of synthetic signals following the method described in [87]. Following the method proposed in [23] to enforce the first and second moments in a white random process, the objective is to obtain a random time series $x(n)$ with given mean μ_x, variance σ_x^2 and autocorrelation function $\rho_x(n)$. The procedure works as follows:

1. A random Gaussian series $G(k)$ [1] with $k = 1, 2, \ldots, N$ is generated in the frequency domain, where N is the length of the desired time series $x(n)$. Every element of $G(k)$ is an independent Gaussian random variable with mean $\mu_G = 0$ and variance $\sigma_G^2 = 1$.

2. The Discrete Fourier Transform (DFT) of the autocorrelation function $\rho_x(n)$ is computed, $S_x(k) = \mathcal{F}[\rho_x(n)]$, where $\mathcal{F}[\cdot]$ is the DFT operator.

3. The entry-wise product $X(k) = G(k) \circ S_x(k)^{\frac{1}{2}}$ is computed.

4. The correlated and Gaussian time series $x(n)$ is obtained as $\mathcal{F}^{-1}[X(k)]$.

Additionally, to emulate the behavior of real world physical signals, some noise can be superimposed to the synthetic signals, so as to mimic random perturbations due to limited precision of the sensing hardware and random fluctuations of the observed physical phenomenon. The noise is commonly modeled as a zero mean white Gaussian process with standard deviation σ_{N}.

4.1.4 Theoretical Bound for Signal Compression

Given a discrete and Gaussian time series $x(n)$, the theoretical lower bound on the transmission rate $R_{\min}$ (bits/sample) can be derived from [11]:

$$R_{\min}(\lambda, N, \varepsilon_T) = \frac{1}{N} \sum_{i=1}^{N} \max \left\{ 0, \frac{1}{2} \log_2 \left(\frac{\zeta_i^2}{\varepsilon_T} \right) \right\}$$

where ε_T is the maximum permitted distortion at the receiver, N is the number of input samples and ζ_i are the eigenvalues of the covariance matrix $\mathbf{\Sigma}(\lambda)$ of $x(n)$ for $n = 1, \ldots, N$. Hence, for given $(\lambda, N, \varepsilon_T)$, the compression ratio achievable by any practical scheme is bound as:

$$\eta \geq \frac{R_{\min}(\lambda, N, \varepsilon_T)}{R_0} \tag{4.4}$$

where R_0 is the rate expressed in bits/sample in the uncompressed case.

---

[1] For an in depth characterization of the Gaussian correlation function, see [4].

4.2 Related work

A comprehensive survey of practical compression schemes for wireless sensor networks can be found in Srisooksai *et al.* [68]. This work classifies a wide selection of existing techniques covering, for each category: performance, open issues, limitations and suitable applications.

More specifically, much of the existing literature has been devoted to the systematic study of lossless compression. In general, these works address the potential benefits of lossless compression in terms of energetic savings, thus providing a useful perception of the trade-offs involved in these strategies. Van Der Byl *et al.* [73] examine Huffman, Run Length Encoding (RLE) and Delta Encoding (DE), comparing the energy spent on compression for these schemes. Liang [42] treats lossy (LTC) as well as lossless (LEC and Lempel-Ziv-Welch) compression methods, but focuses on their compression performance. Along the same lines, Barr and Asanović [8] compare several lossless compression schemes for a StrongArm CPU architecture, showing that data compression in some cases may cause an increase in the overall energy expenditure, which is a particularly relevant result.

The lesson learned is that lossless compression can provide some energy savings. These are, however, smaller than one might expect because, for the sensor hardware in use nowadays, the energy spent on the execution of the compression algorithms may be of the same order of magnitude as that spent on transmission, thus limiting the applicability to energy-constrained wireless devices. This engaging consideration was outlined for first time in [8], and it provided the primary motivation for the present work, thus encouraging us to address the topic in a more systematic way.

Further work has been carried out on lossy compression schemes. Of particular interest for the present chapter, it is worth noting the algorithms of the following: LTC (Lu *et al.* [45]), PLAMLiS (Liu *et al.* [43]) and Enhanced-PLAMLiS (Pham *et al.* [56]), all of which are based on Piecewise Linear Approximation (PLA). Alternative approaches are Adaptive Autoregressive Moving Average (Lu *et al.* [45]), which deal with ARMA models; RACE (Chen *et al.* [17]), which exploits Wavelet-based compression; and a lightweight compression framework based on Differential Pulse Coding Modulation (DPCM) (Marcelloni and Vecchio [47]).

It is remarkable that no systematic energy comparison had previously been carried out for lossy schemes. Many important questions were still open. It was not clear whether lossy compression could be advantageous in terms of energy. Little had been said about the tradeoff between compression ratio and representation accuracy, and how both affect the overall energy expenditure. In addition, it was unclear whether linear and autoregressive schemes could provide any advantages at all when compared to more sophisticated techniques such as Fourier- or Wavelet-based transforms, which have been effectively used to compress audio and video signals, and for which fast and computationally efficient algorithms exist.
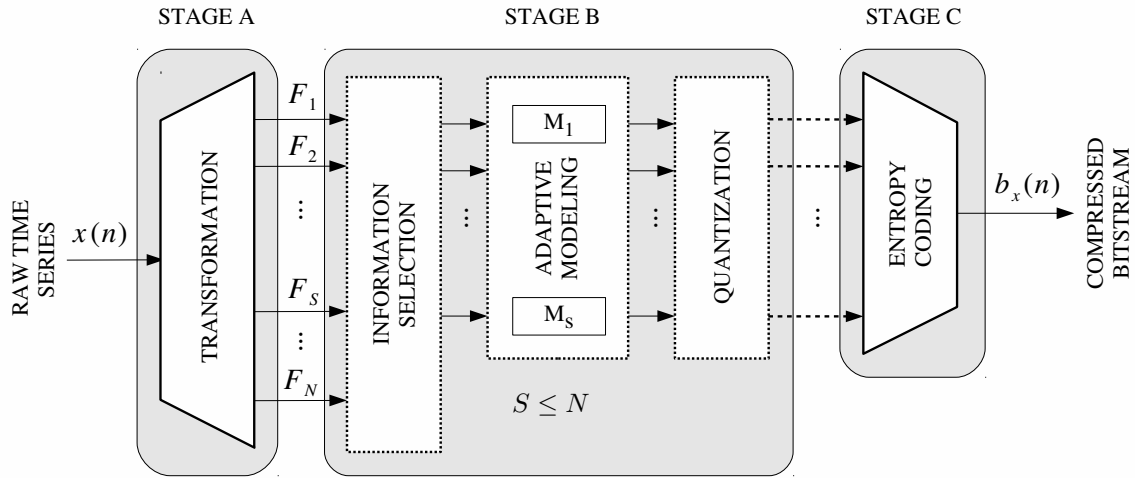
Figure 4.1. *General lossy compression diagram.*

4.3   Lossy Compression Methods for Energy Constrained Devices

In order to facilitate the description of the compression schemes considered in this section and to identify their essential features, Fig. 4.1 shows the diagram of a generic lossy compression algorithm. Following the structure introduced in [80], three fundamental stages are identified:

A **Transformation:** this stage entails the representation of the input signal into a convenient transformation domain. That is, the time series $x(n)$ is decomposed into a number modes, represented by N of coefficients $\{F_1, \ldots, F_N\}$ in the new domain. As an example, FFT, DCT and Wavelet transform time series are decomposed into the frequency domain.

B **Adaptive modeling:** This stage is subdivided into three sub-stages:

   i A set of coefficients $S \leq N$ is selected so that these will be sufficient to represent the signal within a certain target accuracy.

   ii A further adaptive modeling phase (models $\{M_1, \ldots, M_S\}$) can be applied to the time series corresponding to each of the selected coefficients.

   iii Finally, a quantizer can be employed to represent the data through a finite number of levels. This step codifies the level and not the actual value, so that the number of bits for the representation is lower.

This stage is the main source of the information lost in compression. First, the selection stage voluntarily discards part of the information; second, modeling introduces some errors in the representation of each mode; and third, quantization generates additional noise to the signal. Clearly, it is also where most of the compression is achieved.

C **Entropy coding:** the quantized data can be encoded using an *entropy coder* (EC) to obtain additional compression. Entropy represents the amount of information present in the data, and an EC tries to encode the given set of symbols with the minimum number of bits required to represent them.

As a representative example, JPG image compression [80] matches this model as follows:

- Stage-A: DCT transform to frequency domain.

- Stage-B: Quantization for all coefficients, discarding null coefficients after quantization. Run-length encoding for the position of null coefficients. DPCM modeling for the DC coefficients.

- Stage-C: Huffman coding (arithmetic coding is also supported).

It should be noted that a specific compression algorithm does not necessary have to implement all of the above three stages, but some of them can be omitted or only partially taken into account. For example, one could use the selection and quantization blocks in Stage-B, without any adaptive modeling. In the wireless sensors domain, the particular combination of algorithms to use depends on the reconstruction accuracy goal, as well as on the affordable computational complexity.

The next section briefly reviews the lossy signal compression methods that will be characterized in the last section of this chapter. Due to the contained nature of the sensor devices, the selected schemes use only some of the above stages. This is the reason for adopting the following classification criteria. Section 4.3.1 discusses techniques based on Fourier and Wavelet transforms (Stage-A), Section 4.3.2 describes adaptive modeling techniques (Stage-B) and, finally, Section 4.3.3 presents lightweight schemes based on quantization and entropy coding (Stage-C).

4.3.1   Compression Methods Based on Fourier and Wavelet Transforms

Transformation-based techniques achieve compression by sending only selected subsets of the FFT, DCT or Wavelet transformation coefficients. These algorithms first apply an orthogonal mapping of the signal into a suitable domain (Stage-A) and subsequently use the *information selection* block of Stage-B. Among them, the methods presented below differ in how the transformation coefficients are picked. This is not just a technical detail. Throughout this work, the maximum error tolerance must be guaranteed as a system (application) level specification. This approach requires a specific treatment of this family of algorithms, which contrasts with the classic methods in audio and image compression, where reconstruction accuracy is not a hard constraint but a performance metric.

Fast Fourier Transform

The first considered method relies on the simplest way to use Fourier transforms for compression. Specifically, the input time series $x(n)$ is mapped to its frequency representation $X(f) \in \mathbb{C}$ through a Fast Fourier Transform (FFT). As the FFT output domain is the complex plane, it is convenient to define $X_{\mathcal{R}}(f) \doteq \mathfrak{Re}\{X(f)\}$ and $X_{\mathcal{I}}(f) \doteq \mathfrak{Im}\{X(f)\}$ as the real and the imaginary part of $X(f)$, respectively. Since $x(n)$ is a real-valued time series, $X(f)$ is Hermitian, i.e., $X(-f) = \overline{X(f)}$. This symmetry allows the FFT to be stored using the same number of samples N of the original signal. For N even it is enough to take $f \in \{f_1, \ldots, f_{N/2}\}$ for both $X_{\mathcal{R}}(\cdot)$ and $X_{\mathcal{I}}(\cdot)$; while, if N is odd, the required selection is $f \in \{f_1, \ldots, f_{\lfloor N/2 \rfloor + 1}\}$ for the real part and $f \in \{f_1, \ldots, f_{\lfloor N/2 \rfloor}\}$ for the imaginary part. The compressed representation $\hat{X}(f) \doteq \hat{X}_{\mathcal{R}}(f) + j\hat{X}_{\mathcal{I}}(f)$ will also be in the frequency domain and it is built (for the case of N even) as follows:

1. initialize $\hat{X}_{\mathcal{R}}(f) = 0$ and $\hat{X}_{\mathcal{I}}(f) = 0, \forall f \in \{f_1, \ldots, f_{N/2}\}$;

2. select the coefficient with max. absolute value from $X_{\mathcal{R}}$ and $X_{\mathcal{I}}$, i.e., $f_{\max} \doteq \mathrm{argmax}_f\{|X_{\mathcal{R},\mathcal{I}}(f)|\}$ and $M \doteq \mathrm{argmax}_{i \in \{\mathcal{R}, \mathcal{I}\}}\{|X_i(f_{\max})|\}$;

3. set $\hat{X}_M(f_{\max}) = X_M(f_{\max})$ and then set $X_M(f_{\max}) = 0$;

4. if $\hat{x}(n)$ (the inverse FFT of $\hat{X}(f)$) meets the error tolerance constraint, then continue; otherwise repeat from step (2);

5. encode the values and the positions of the harmonics stored in $\hat{X}_{\mathcal{R}}$ and $\hat{X}_{\mathcal{I}}$.

Hence, the decompressor at the receiver obtains $\hat{X}_{\mathcal{R}}(f)$ and $\hat{X}_{\mathcal{I}}(f)$, and it exploits the Hermitian symmetry to reconstruct $\hat{X}(f)$. Note that the above coefficient selection method resembles a K non-linear approximation, which is usually implemented by image processing techniques (see, e.g., [29]), in the sense that it uses the magnitude as a criterion to select the harmonics. For a time series, K (the number of coefficients to be retained) is dynamically selected depending on the input signal characteristics.

Also note that many alternative schemes for the selection of the Fourier coefficients are possible. For instance, one may select the FFT coefficients based on the maximum absolute magnitude of their complex values and then retain both the real and imaginary part of the selected coefficients. Nevertheless, differences among the various approaches are insignificant.

Low-Pass Filter Fast Fourier Transform

The *Low-Pass Filter* (LPF) is a FFT-based lossy algorithm alternative to the generic method. Since the input time series $x(n)$ is a slowly varying signal in many common cases (i.e., it has a strong temporal correlation), with some high frequency noise superimposed, most of the significant coefficients

of $X(f)$ reside in the low frequencies. FFT-LPF is initialized to $\hat{X}(f) = 0$ for all frequencies. Thus, $X(f)$ is evaluated from f_1, incrementally moving toward higher frequencies, $f_2, f_3, \ldots$. At each iteration i, $X(f_i)$ is copied onto $\hat{X}(f_i)$ (both the real and imaginary parts), the inverse FFT is computed taking $\hat{X}(f)$ as input, and the error tolerance constraint is checked on the thus obtained $\hat{x}(n)$. If the given tolerance is met, the algorithm stops; otherwise it is reiterated for the next frequency f_{i+1}.

Note that this method resembles a K linear approximation scheme in which the selection order is fixed (LPF). Furthermore, the number of coefficients to be retained, K, is dynamically adjusted in order to meet a given error tolerance. In general terms, this low-pass approach for the coefficients selection is similar to several audio compression algorithms, in that it retains the harmonics according to its frequency.

Windowing

The two algorithms discussed above suffer from an edge discontinuity problem. In particular, when computing an FFT over a window of N samples, if $x(1)$ and $x(N)$ differ substantially the information about this discontinuity is spread across the whole spectrum in the frequency domain. Hence, in order to meet the tolerance constraint for all the samples in the window, a high number of harmonics is selected by the previous algorithms, resulting in poor compression and in a high number of operations.

To solve this issue, a refined version of the FFT algorithm considers overlapping windows of $N + 2W$ samples (instead of disjoint windows of length N), where W is the number of samples that overlap between subsequent windows. The first FFT is taken over the entire window and the selection of the coefficients goes on, depending on the selected algorithm (either FFT or FFT-LPF), but the tolerance constraint is only checked on the N samples in the central part of the window. With this workaround, it is possible to get rid of the edge discontinuity problem and encode the information about the N samples of interest with very few coefficients, as will be shown shortly in Section 4.4. As a drawback, the direct and inverse transforms have to be taken on longer windows, which results in a higher number of operations.

Discrete Cosine Transform (DCT)

Discrete Cosine Transform (type II) is included in this study mainly for the following three reasons. First, its coefficients are real. Coping with real and imaginary parts is not required, and thus we save both memory and computation. Second, it has a strong *energy compaction* property [61]. In other words, most of DCT's signal information tends to be concentrated in a few low-frequency components. And, third, the DCT of a signal with N samples is equivalent to the DFT of a real signal with even symmetry and double length; so the DCT does not suffer from the edge discontinuity problem.

Wavelet Transforms

As an alternative to Fourier schemes, several methods based upon multi-resolution analysis have been proposed in the literature. RACE [17] is a notable example: it features a compression algorithm based on the Fast Wavelet Transform (FWT) of the signal (Stage-A), followed by the selection of a number of coefficients (Stage-B) that are used to represent the input signal within given error bounds. As for DCT schemes, the compression takes place mainly in the selection step.

In [17], a Haar basis function is used for the wavelet decomposition step. The most remarkable contribution of RACE is the way in which the wavelet coefficients are selected. Most traditional compression algorithms follow the FWT methodology by simply picking the largest coefficients. In other words, the selection step is based on a threshold value: all the coefficients below the threshold are discarded, whereas those above it are retained. What is different about RACE is that the Haar wavelet coefficients are arranged into a tree structure. Then, thanks to some special properties of the Haar functions, the error in the reconstruction of the signal is estimated at each node of the tree, assuming that this node (i.e., the corresponding coefficient) and all its children in the tree are omitted.

This selection method has two important properties. First, the signal representation error can be evaluated on-the-fly during the decomposition, and the maximum error tolerance can be kept under control without having to compute any inverse wavelet transform. Note that in the FFT and DCT methods above, the error tolerance check always entails the computation of an inverse transformation at the source. Second, compression can be achieved in an incremental way by descending the tree and adding nodes until the desired precision is reached (of course, the higher the number of coefficients, the lower the compression performance). These facts are very important for energy-constrained WSNs and, as presented in Section 4.4, they lead to less energy being allocated to compression when compared to the DCT and FFT schemes.

4.3.2   Compression Methods Based on Adaptive Modeling

In *Adaptive Modeling* schemes, some signal model is iteratively updated over time, exploiting the correlation structure of the signal through linear, polynomial or autoregressive methods. Specifically, the input time series is collected and processed according to transmission windows of N samples each. At the end of each time window, the selected modeling method is applied. The result is a set of model parameters that are transmitted in place of the original data, with the expectation that the number of model parameters is lower than the number of samples in the original window. Note that information selection is not used in the adaptive modeling schemes described below, as they do not employ any transformation stage.
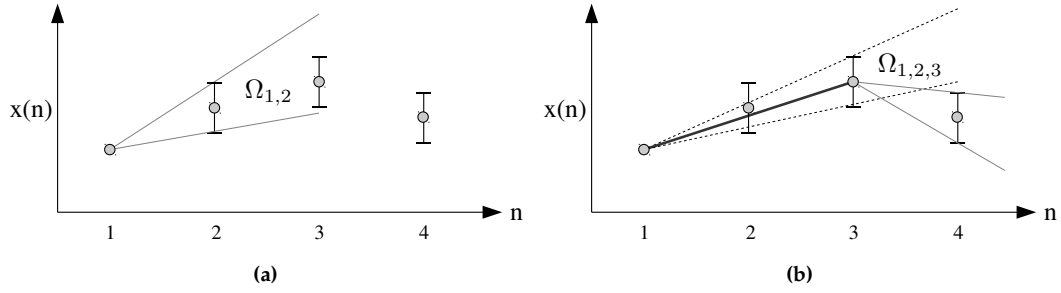
Figure 4.2. *Lightweight Temporal Compression example: first line segment.*

Piecewise Linear Approximations

The idea behind *Piecewise Linear Approximations* (PLA) is to use a sequence of line segments as models to represent an input time series $x(n)$ over pre-determined time windows of N samples, with a bounded approximation error. For slowly varying time series (such as most environmental measures), linear approximations work well enough over short time frames. Further, since a line segment can be determined by only two end points, PLA leads to quite efficient implementations in terms of memory and transmission requirements.

Most PLA algorithms use standard least squares fitting to calculate the approximating line segments (hereafter referred to as $\hat{x}(n)$). Often, a further simplification is introduced to reduce the computational complexity, which consists of forcing the end points of each line segment to be points of the original time series $x(n)$. This makes least squares fitting unnecessary, as the line segments are fully identified by the extreme points of $x(n)$ in the considered time window. The following schemes exploit this approach.

Lightweight Temporal Compression (LTC) [64]: the LTC algorithm is an efficient, low complexity PLA technique. LTC works as follows. Let $x(n)$ be the points of a time series with $n = 1, 2, \ldots, N$. The LTC algorithm starts with $n = 1$ and fixes the first point of the approximating line segment to $x(1)$. The second point $x(2)$ is transformed into a vertical line segment that determines the set of all acceptable lines $\Omega_{1,2}$ with starting point $x(1)$. This vertical segment is centered at $x(2)$ and covers all values meeting a maximum tolerance ε_T, i.e., those that lay within the interval $[x(2) - \varepsilon_T, x(2) + \varepsilon_T]$ (see Fig. 4.2a). The set of acceptable lines for $n = 3$, $\Omega_{1,2,3}$, is obtained by the intersection of $\Omega_{1,2}$ with the set of lines that have starting point $x(1)$ and that are acceptable for $x(3)$ (dashed lines in Fig. 4.2b). The procedure is iterated by adding one point at a time until, at a given step s, some $x(i)$ is not contained in $\Omega_{1,2,\ldots,S}$. Thus, the algorithm sets $x(1)$ and $x(s-1)$ as the starting and ending points of the approximating line segment for $n = 1, 2, \ldots, s-1$ (solid line in Fig. 4.2b). It then starts over with $x(s-1)$, considering it to be the first point of the next approximating line segment. In this example, $s = 4$.

PLAMLiS [43]: Similarly to LTC, *Piecewise Linear Approximation with Minimum number of Line Segments* represents the input data series $x(n)$ through a sequence of line segments. Here, the linear fitting problem is converted into a set-covering problem, trying to find the minimum number of segments that cover the entire set of values over a given time window. This problem is solved through a Greedy algorithm as explained in [43]. This algorithm is outperformed in terms of complexity by E-PLAMLiS.

(a) *First segment.*   **(b)** *Partition in two segments.*
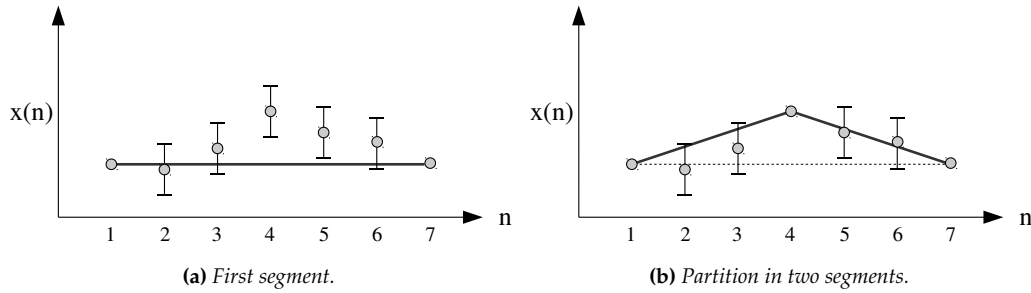
Figure 4.3. *Enhanced PLAMLiS example: first segment splitting.*

Enhanced PLAMLiS [56]: This is a top-down recursive segmentation algorithm with smaller computational costs in comparison to PLAMLiS. Consider the input time series $x(n)$ and a time window $n = 1, 2, \ldots, N$. The algorithm starts by taking a first segment $(x(1), x(N))$ (Fig. 4.3a). If the maximum allowed tolerance ε_T is met for all points along this segment, the algorithm ends. Otherwise, the segment is split into two segments at the point $x(i)$, $1 < i < N$, where the error is maximum, and it obtains the two segments $(x(1), x(i))$ and $(x(i), x(N))$ ($i = 4$ in Fig. 4.3b). The same procedure is recursively applied to the resulting segments until the maximum error tolerance is met for all points.

Polynomial Regression

The above methods can be modified by relaxing the constraint that the endpoints of the segments $x(i)$ and $x(j)$ ($j > i$) must be actual points of $x(n)$. In this case, polynomials of given order $p \geq 1$ are used as the approximating functions, whose coefficients are found through standard regression methods based on least squares fitting [57]. Specifically, the *Polynomial Regression* (PR) algorithm starts with a window of p samples (since a p-order polynomial exactly interpolates p points), for which the best fitting polynomial coefficients are obtained. Thus, the algorithm keeps increasing the window length of one sample at a time, computing the new coefficients, and it stops when the target error tolerance is no longer met.

Intuitively, there is very low computational complexity involved in tracing a line between two fixed points, as done by LTC and PLAMLiS, while least squares fitting may have a significant cost. Then, polynomial regression can obtain better results in terms of approximation at the cost of higher computational complexity, which increases with the polynomial order.

Autoregressive Methods

Autoregressive (AR) models in their multiple flavors (AR, ARMA, ARIMA, etc.) have been widely used for time series modeling and forecasting in fields like macro-economics or market analysis. The basic idea is to obtain a model based on the history of the sampled data, i.e., on its correlation structure. When used for signal compression, AR obtains a model from the input data and sends this model to the receiver in place of the actual time series. The reconstructed model is thus used at the data collection point (the sink) for data prediction until it is updated by the encoder device. Specifically, each node locally verifies the accuracy of the predicted data values with respect to the collected samples. If the accuracy is within a prescribed error tolerance, the node assumes that the current model will be sufficient for the sink to rebuild the data within the given error tolerance (Fig. 4.4a). Otherwise, the parameters from the current model are encoded, and a new model is built as a replacement for the old one (Fig. 4.4b). As stated above, the model parameters are sent to the sink at the end of each transmission window in place of the original data.

(a) *First model.* **(b)** *Model update.*
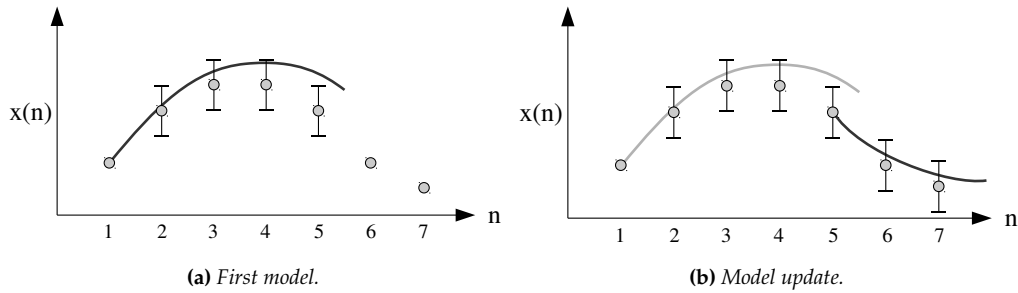
Figure 4.4. *Autoregressive Models.*

Adaptive Autoregressive Moving Average (A-ARMA) [45]: the basic idea of A-ARMA [45] is that of having each sensor node compute an ARMA model based on $N' < N$ consecutive samples. In order to reduce the complexity in the model estimation process, adaptive ARMA employs low-order models, whereby the validity of the model being used is checked through a moving window technique. Specifically, a sensor node builds an ARMA model $M^{(0)} = ARMA(p, q, N', 0)$ considering N' samples starting from the first sample (sample 0) of the current transmission window (p and q are the orders related to the autoregressive and moving average components of the ARMA filter). Hence, this model is updated considering N' subsequent samples at a time until the prescribed error tolerance is met, at which point a new ARMA model is built and the update/check procedure is iterated for this one. At the end of the transmission window of N samples, the parameters of all the ARMA models that have been obtained to describe the input time series (within the prescribed error tolerance) are sent to the sink in place of the original data, as discussed above.

Modified Adaptive Autoregressive (MA-AR): according to A-ARMA, the model is updated over fixed-size windows of N' samples. A drawback of this is that the estimation over fixed-size windows can lead to poor results when used for forecasting, especially for highly noisy environments. MA-AR allows the estimation to be performed on time windows whose size is adapted according to the signal statistics. A more detailed discussion of ARMA methods can be found in [87].

4.3.3   Compression Methods Based on Entropy Coding

This work adopts the algorithm proposed in [47] (MV) as the representative of Stage-C. This algorithm consists of three steps: (a) Differential Pulse-Modulation Coding (DPCM), (b) quantization and (c) Huffman entropy encoding.

After de-noising, step (a) employs a simple differential encoding model (DPCM), which operates on the differences between consecutive input samples. The rationale behind this differential scheme is that WSN signals are usually smooth and slow time-varying. Hence, the difference between samples is expected to be small, leading to a small amount of information to be encoded. This technique is broadly used, as in the already mentioned codification of JPG coefficients.

In the quantization block (b), the difference between subsequent samples is quantized. As stated before, this is the step of the algorithm where most of the compression performance is achieved. Indeed, given the small expected value of the DPCM differences, a quantizer with only a small number of levels can be used without to much impact on the signal representation's accuracy.

In the performance evaluation, the maximum error tolerance is bound for each sample, setting it as a constant input parameter common for all algorithms analyzed. Therefore, the DPCM algorithm has been adapted to consider this. Specifically, a first pass is performed to find the maximum difference at the output of the DPCM. Based on this, the number of levels of the quantizer is selected so that the quantization error remains smaller than a target error tolerance; this returns the quantizer for the given input signal. After this, a second pass is executed using the selected quantizer to obtain the final encoded symbols. Note that this is slightly different from [47], where optimal quantizers are calculated offline through a dedicated optimization stage that follows different optimization criteria. While the latter approach is also valuable, it does not allow for precise control of the maximum error tolerance and a fair comparison with the other compression schemes.

Finally, the entropy encoding step (c) exploits the fact that the quantization levels have different probabilities. Hence, a Huffman encoder is designed to assign the shorter binary codewords to the most probable levels. This dictionary can be sent together with the compressed data frame, or it can be statistically precomputed and shared between the communicating entities.

4.4 Performance Analysis

The goal of this section is to quantify the energy gains that may be obtained through the use of compression and to evaluate the impact on reconstruction accuracy. Even more interesting, the ultimate aim would be to predict the reconstruction error for a given energy constraint. More specifically, when the total energy is bounded, the compression ratio can be adjusted to meet specifications. Yet, this implies sacrificing some reconstruction accuracy, which would be desirable to predict in advance. With this focus in mind, this section deals with the following particular objectives:

- To provide a thorough performance comparison of the compression methods of Section 4.3. The selected performance metrics are: 1) compression ratio, 2) computational and transmission energy and 3) reconstruction error at the receiver.

- To explore and quantify the impact on the compression performance of the statistical properties of the input signals.

- To assess whether or not data compression leads to energy savings in single- and multi-hop networks, establish the conditions required, and obtain quantitative measurements of possible benefits as a function of compression ratio and energy consumption of the device hardware.

Toward the above objectives, a first set results is obtained by simulation using synthetic signals with a well characterized correlation length. These signals make it possible to provide a fine grained description of the performance of the selected techniques, so as to look comprehensively at the entire range of variation in the temporal correlation statistics. Real datasets are then used to validate the obtained fitting results.

This section focuses on single- and multi-hop WSNs, where the interference due to channel access is negligible or absent. In this case, the energy expenditure at the MAC is only confined to transmission and reception energy by also keeping into account the protocol overhead at the MAC in terms of packet headers. However, further energetic inefficiencies due to channel contentions are neglected, as well as waiting times due to floor acquisition.

4.4.1 Experimental Setup

The platform selected for this study is based on the TI MSP430 [12] micro-controller. A specific 16-bit floating point package for this architecture was used in arithmetic operations and data representation. The number of clock cycles needed for the floating point operations running on this architecture are given in Table 5.8 of [12]. In the active state, the MSP430 is powered by a current of $330\,\mu$A at 2.2 V, and it has a clock rate of 1 MHz. The resulting energy consumption per CPU cycle is $E_I = 0.726$ nJ.

The platform includes the TI CC2420 RF transceiver [20], an IEEE 802.15.4 [2] compliant radio. For commercial radio transceivers, the current consumption associated with the transmission activity is typically selected from a finite set of values (See section 2.3.1 and, in particular, Eq. (2.7)). The CC2420 has 8 levels, varying from a minimum of 8.5 mA to a maximum of 17.4 mA, with a supply voltage of 3.3 V for an effective data rate of 250 kbps (see [20]). Thus, given the current power level $\ell \in \{1, \ldots, 8\}$, the cost associated with the transmission of a bit, $E'_T[\ell]$, ranges from 112 nJ to 230 nJ, respectively. Note that the current level, and consequently the output power of the radio transceiver, has to be chosen according to the considered scenario, which includes the transmission distance, the channel noise level, the type of environment (e.g., free space, indoor, presence of obstacles), etc.

The results obtained for this specific architecture can be promptly generalized to different CPUs and radios. As explained in Section 2.5.1, this is made possible by separating algorithm-dependent and hardware-dependent terms in the calculation of the overall energy consumption. For this platform, the energy required corresponds to the energy spent by the micro-processor during 154 and 316 clock cycles, depending on the radio power selected. It is interesting to compare this value with results found for the Strong-ARM architecture [8].

4.4.2   Simulation Setup

For every compression method, the number of operations to process the original time series $x(n)$ is recorded, accounting for the number of additions, multiplications, divisions and comparisons. For selected hardware architecture, the number of operations is mapped into the corresponding number of clock cycles, and the latter is subsequently mapped into the energy expenditure. The computation only accounts for the CPU operations that are related to compression algorithms.

Regarding communication costs, only the transmission energy is taken into consideration while neglecting the cost of switching the radio transceiver on and off and the energy spent at the destination to receive the data. The former are fixed costs that would also be incurred without compression, while the latter can be ignored if the receiver is not a power-constrained device. Moreover, link-level retransmissions due to channel errors or multi-user interference are not considered.

The set of test signals is generated by following the method described in Section 4.1.3. Specifically, the method makes use of a Gaussian correlation function [4], i.e., $\rho_x(n) = \exp\{-an^2\}$, where a is chosen in order to get the desired correlation length λ, as follows:

$$a = -\frac{\log(\rho_{\text{th}})}{(\lambda)^2}$$

Without loss of generality, synthetic signals are generated with $\mu_x = 0$ and $\sigma_x^2 = 1$. In fact, applying an offset to the generated signals and a scale factor does not change the resulting correlation. Further details of the Gaussian correlation function can be found in [4].

In this section, all results are obtained with the following simulation settings:

- The length of the time series is $N = 500$ samples (time slots) at a time, progressively taken from a longer realization of the signal, so as to avoid artifacts related to the generation technique. For a fair comparison, the same realization of the input time series $x(n)$ has been used for: all the compression methods considered, each simulation run and the value of the correlation length.

- The correlation length of synthetic signals is set to $\lambda \in \{1, 10, 20, 50, \ldots, 500\}$ samples, where after 20, λ varies in steps of 30, with $\rho_{\text{th}} = 0.05$ (See Eq. (4.3)).

- A Gaussian noise with standard deviation $\sigma_{\text{N}} = 0.04$ has been added to the signal to emulate random noise in the sensing process.

- For reconstruction accuracy, the absolute error tolerance has been set to $\varepsilon_T = \xi\sigma_N$, with $\xi \geq 0$. Moreover, all the compression algorithms have been configured with the same error tolerance, so that the energy compression and consumption figures we obtain are for the same reconstruction fidelity at the receiver.

- Each point is obtained by averaging the outcomes of 10^4 simulation runs.

4.4.3 Compression Ratio Performance

This section analyzes the performance in terms of compression and compression energy (the latter directly related to computational complexity) for the lossy compression methods presented in Section 4.3. This section should allow discarding some methods that are prohibitively expensive in terms of computation, and hence they are not suited for low-power wireless applications. The next section will go in depth into the analysis of those methods that prove to be more efficient.

Adaptive Modeling Methods

The first set of tests compares the performance of the following compression methods:

- Modified Adaptive Autoregressive (M-AAR) for two orders, $p = \{2, 4\}$;

- Polynomial Regression (PR) for two orders, $p = \{2, 4\}$;

- Piecewise Linear Approximation (PLAMLiS) and its Enhanced version (E-PLAMLiS);

- Lightweight Temporal Compression (LTC)
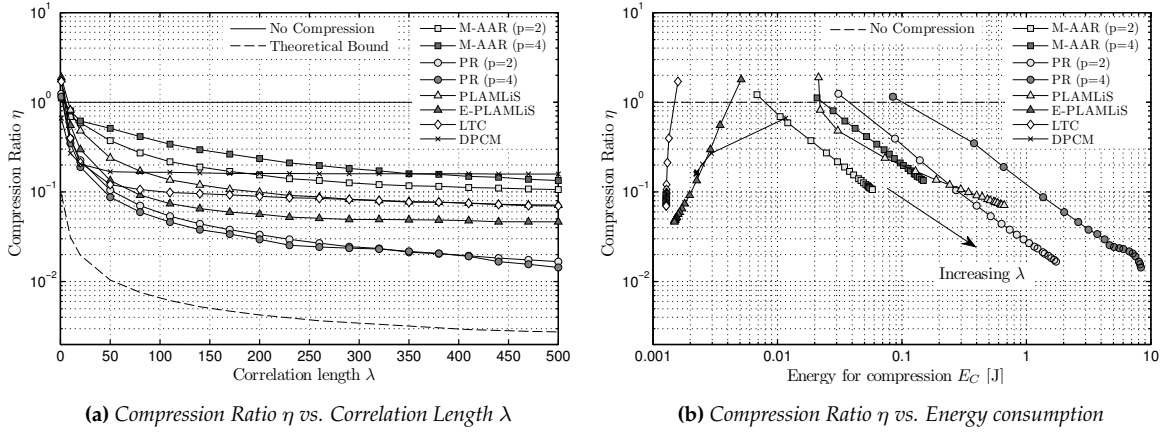
- Modified DPCM algorithm (DPCM).

(a) *Compression Ratio η vs. Correlation Length λ*

(b) *Compression Ratio η vs. Energy consumption*

Figure 4.5. *Compression performance for the Adaptive Modeling methods for fixed $\varepsilon_T = 4\sigma_n$.*

Fig. 4.5a shows the Compression Ratio achieved by the six compression methods as a function of the correlation length λ. The lower bound on the compression ratio η is also plotted for comparison (see Section 4.1.4). The high correlation between λ and η, shown in this plot, indicates that λ is in fact capturing the information, and confirms that it is the right choice for parameterizing the performance of all compression schemes.

These results reveal that for small values of λ the compression performance is poor for all compression schemes, whereas it improves for increasing correlation length and, finally, it reaches a floor value for sufficiently large λ. Also, the compression performance differs among the different methods, with PR giving the best results. This reflects the fact that, differently from all the other methods, PR approximates $x(n)$ without requiring its fitting curves to pass through the points of the given input signal. This entails some inherent filtering, which is embedded in this scheme and makes it more robust against small and random perturbations.

Fig. 4.5b shows the relation between the energy consumption for compression E_C and the compression ratio η. For increasing values of λ, the compression ratio becomes better for all schemes, but their energy expenditure differs substantially. Notably, the excellent compression capabilities of PR are counterbalanced by its demanding requirements in terms of energy. M-AAR and PLAMLiS also require quite a large amount of processing energy, although this is almost one order of magnitude smaller than that of PR. Finally, LTC and E-PLAMLiS have the smallest energy consumption among all schemes.

It is interesting to discuss the dependence of the energy spent on compression (which is directly related to the computational complexity) of λ. LTC encodes the input signal $x(n)$ incrementally, starting from the first sample and adding one sample at a time. Thus, the number of operations that it performs depends only weakly on the correlation length. In turn, the energy that it spends

on compression is almost constant with varying λ. E-PLAMLiS takes advantage of the increasing correlation length: as the temporal correlation increases, this method has to perform fewer "divide and reiterate" steps, so the number of operations required gets smaller and, consequently, the energy spent on compression is also reduced. DPCM performs almost the same number of operations for different correlation lengths, except for very small values of λ. This occurs because, in order to meet the error constraint for uncorrelated signals ($\lambda \approx 1$), the quantization step has to use a high number of levels (DPCM signals have wider ranges); and with an increasing number of levels, the entropy encoder assigns an exponentially increasing number of bits to some symbols. As a consequence, the number of operations related to the assignment of these codewords increases.

For the remaining methods, the complexity grows with λ. For PLAMLiS, this is due to the first step of the algorithm, where for each point the longest segment that meets the given error tolerance has to be found (see Section 4.3). When $x(n)$ is highly correlated, these segments become longer and PLAMLiS has to check the tolerance constraint a large number of times for each of the N samples of $x(n)$. For M-AAR and PR, every time a new sample is added to a model (autoregressive for the former and polynomial for the latter), this model must be updated and the error tolerance constraint has to be checked. These tasks have a complexity that grows with the square of the length of the current model. Increasing the correlation length of the input time series also increases the length of the models, leading to better compression ratios but, in turn, higher energy consumption.

Fourier- and Wavelet-based Methods

The second set of results compares the performance of the Fourier- and Wavelet-based compression schemes of Section 4.3, considering the same simulation setup as above.

Fig. 4.6a shows that the compression performance of Fourier-based methods still improves with increasing λ. The methods that perform best are FFT Windowed, FFT-LPF Windowed and DCT-LPF, all of which achieve optimal compression ratios. Notably, η is around 10^{-2} for $\lambda \geq 300$ (A compression factor of more than $100\,\text{X}$). Conversely, FFT and FFT-LPF need to encode more coefficients to meet the prescribed error tolerance constraint, due to their edge discontinuity problem, and thus their compression ratio is worse. RACE is outperformed by other DCT-based solutions in terms of compression performance at all correlation lengths. As will be discussed shortly, this scheme may be interesting for its lightweight character in terms of energy consumption requirements.

The energy cost for compression is reported in Fig. 4.6b, where λ is varied as an independent parameter. The compression costs for all the FFT/DCT schemes consist of a fixed contribution, which represents the energy needed to evaluate the FFT/DCT of the input signal $x(n)$. Thus, there is a second contribution which depends on the number of transformation coefficients that are picked. Specifically, a decreasing λ means that the signal is less correlated and, in this case, more coefficients are to be considered in order to meet a given error tolerance. Further, for each of them, an inverse
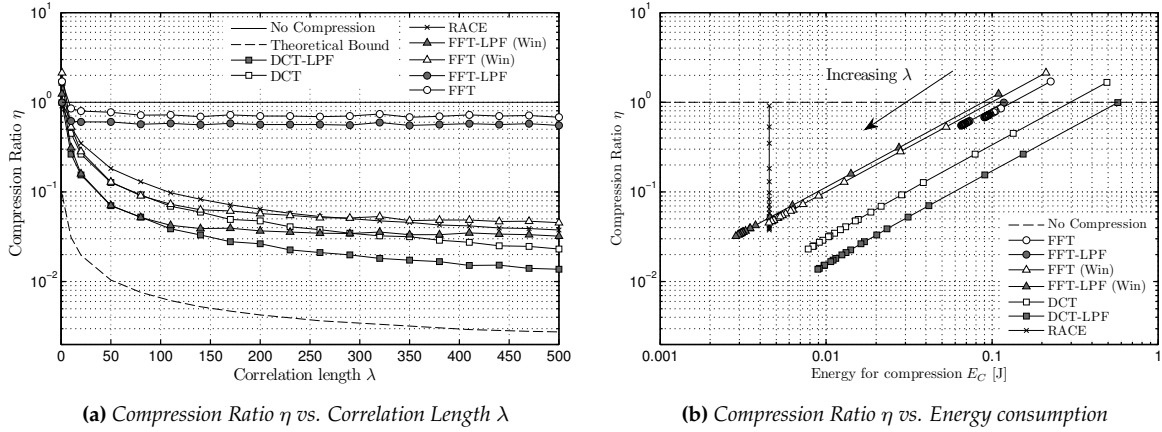
(a) *Compression Ratio η vs. Correlation Length λ*

(b) *Compression Ratio η vs. Energy consumption*

Figure 4.6. *Compression performance for the Fourier-based methods for fixed* $\varepsilon = 4\sigma_n$.

transform has to be evaluated to check whether an additional coefficient is required. This leads to an increasing computational cost for decreasing λ.

Instead, RACE performs only an initial Wavelet decomposition. Then it subsequently checks the reconstruction error –thanks to the coefficient selection phase along the constructed tree– without having to compute an inverse transform at each step. Hence, its energy consumption remains nearly constant for different correlation lengths λ, making the consumption significantly lower than that of the FFT and DCT schemes.

Summarizing the results, FFT-based methods achieve the best compression ratio performance among all the schemes in Figs. 4.5b and 4.6b (DCT-LPF is the best performing algorithm), whereas PLA schemes give the best energy consumption performance for compression (LTC is the best among them).

Applicability to real-world signals.

Correlation length λ is still somehow an abstract parameter. It is interesting to evaluate the correlation length of different real-world signals in order to get an idea of the typical orders of magnitude involved. Table 4.1 shows the typical sampling rate and the correlation length for selected real-world signals. Luminosity and temperature data are taken from the database used in [60]; readings from load sensors are taken from a structural-monitoring WSN installed by WorldSensing in the Palau Sant Jordi of Barcelona (ES); and seismic data is obtained from the measurements in [74]. The high quality (HQ) musical sample and speech data are, respectively, from an excerpt of classical music by Mozart and a sample of speech from an adult female; these datasets are available at [30]. The low quality (LQ) musical sample is from the Händel's Messiah: Hallelujah Chorus.

This work focuses on compression schemes that are signal-agnostic and, as such, try to approxi-

Signal type	Sampling rate [Hz]	Typical λ [samples]
Indoor temperature	1/60	563
Humidity	1/600	355
Load sensors	1/5	402
Outdoor temperature	10	135
Luminosity	1/300	100
Music (HQ)	44.1 k	33
Music (LQ)	8192	4
Speech	8192	8
Seismic	150	3

Table 4.1. *Typical correlation length λ for selected real-world signals.*

mate the signals on the fly through some modeling technique. However, these techniques are effective only for slowly varying signals with, say, correlation lengths larger than 50 samples. Typically, this is the case for many signals monitored by WSNs gathering climatic/environmental data or structural health (first section of Table 4.1). Audio signals, such as music and voice, seismic signals, or signals related to online traffic monitoring show abrupt variations with high non-stationarity. For this reason, these signals are characterized by very short correlation lengths, usually smaller than 10 samples (second half of Table 4.1).

While the techniques presented here can be used for the compression of these specific signals, dedicated algorithms are expected to lead to better results (JPG, MP3, MPG, etc.). However, these highly-customized algorithms are not suitable for the typical WSN approach, which is characterized by the low-energy requirements of the devices and the broad range of applications.

4.4.4   Energy Performance of Compression Algorithms

This section examines the selected compression methods by considering the total energy E for a single- and multi-hop scenarios. E_{CT} accounts for compression E_C and transmission E_T. So now both the compression ratio η (for transmission energy) and computational complexity (for transmission energy) affect to the performance, and hence should be evaluated together.

Single-hop Performance

Fig. 4.7a shows the performance in terms of compression ratio η *vs.* total energy consumption E for a set of compression methods when applied to a single-hop scenario. PLAMLiS was not considered, as its performance is always dominated by E-PLAMLiS, and we show the performance of only the best Fourier-based schemes. The cross marks the case where no compression is applied to the signal, which is sent entirely to the gathering node. Note that energy savings can only be obtained for those cases where the total energy lies to the left of the raw transmission (Vertical dashed line).
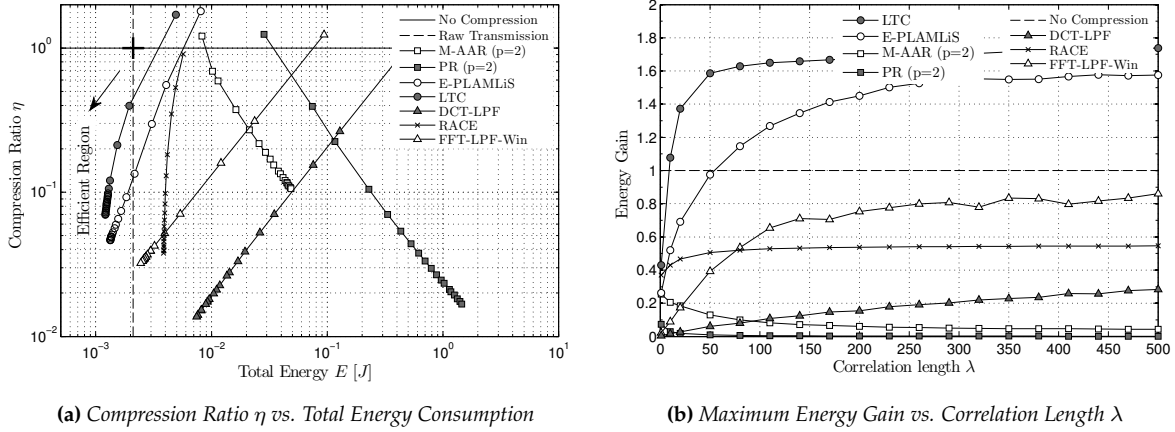
(a) *Compression Ratio η vs. Total Energy Consumption*

(b) *Maximum Energy Gain vs. Correlation Length λ*

Figure 4.7. *Energy efficiency. Comparison of lossy compression schemes for a single-hop scenario.*

For the following results, the transmission power of the radio transceiver is set to the maximum level, in order to show the best achievable performance when data compression is applied. Despite adopting the maximum power level, the computational energy is notably comparable to that spent on transmission; thus, only LTC and Enhanced PLAMLiS can achieve some energy savings. All the other compression methods entail a high number of operations and, in turn, perform worse than the no compression case in terms of overall energy expenditure.

This remarkable result is a consequence of the fact that –among the selected technologies– only a few hundred CPU instructions can be executed to compress a single bit of information while being energetically efficient (See Section 4.4.1). Obviously, the ratio between radio and processor consumption in Eq. (4.2) can be quite different among platforms, especially for other radio technologies; but the obtained results indicate that this ratio should be carefully evaluated.

Fig. 4.7b shows the total energy gain, defined as the ratio between the energy spent on transmission (in the case with no compression) and the total energy spent on compression and transmission using the selected compression techniques (see Section 4.1.2). The method that offers the highest energy gain is LTC, although other methods such as DCT-LPF can achieve better compression performance (see Fig. 4.7a). Note that in this scenario the total energy is highly influenced by the computational cost (even for the maximum output power for the radio). Thus, the most lightweight methods, such as LTC and enhanced PLAMLiS, perform best.

Table 4.2 qualitatively summarizes the performance of the considered signal compression algorithms, classifying them in terms of compression capabilities, energy requirements (directly related to their computational complexity) and dependence on the temporal correlation length λ. The third column shows the minimum λ required for an algorithm to be efficient (See Fig. 4.7b). It is interesting to compare this value with the typical λ of real world signals in Table 4.1.

Table 4.2. *Summary of performance for the considered compression methods.*

Compression Method	Compression capabilities	Energy Requirements	Energy Efficiency†	Complexity versus λ
PLAMLiS	average	high	$\times$	increasing
E-PLAMLiS	average	low	10	decreasing
LTC	average	low	50	decreasing
PR	very high	very high	$\times$	increasing
M-AAR	low	high	$\times$	increasing
MV	low	moderate	$\times$	decreasing
FFT	low	very high	$\times$	decreasing
FFT-LPF	low	very high	$\times$	decreasing
FFT Win	high	high	$\times$	decreasing
FFT-LPF Win	very high	high	$\times$	decreasing
DCT	high	high	$\times$	decreasing
DCT-LPF	very high	high	$\times$	decreasing
RACE	average/high	moderate	$\times$	constant

† Minimum required correlation length λ

Multi-hop Performance

In a multi-hop scenario, the situation can be quite different. The extra processing cost required for more compression can be balanced because a shorter message is relayed several times to reach the data gathering point. This section investigates whether further gains are possible when the compressed information has to travel multiple hops.

In this case, both transmission and reception energy are accounted for at each intermediate relay node. This implies that, for an intermediate node, the amount of energy saved is roughly twice that of a terminal sensor. Additionally, nodes close to the sink tend to aggregate the traffic of multiple children, so energy savings for these nodes are especially important for the network's survival.

In the following, only LTC and DCT-LPF are discussed, as these are the two methods that perform best in terms of complexity and compression efficiency, respectively. Obviously, the study can be extended to any other method.

In Fig. 4.8a, the results for DCT-LPF are shown with black filled markers, whereas white filled markers are used for LTC. The correlation length of the input signal is set to $\lambda \in \{300, 500\}$. The type of marker indicates the correlation length of the input signal, specifically: ($\square$, $\blacksquare$) for $\lambda = 300$, ($\circ$, $\bullet$) for $\lambda = 500$. The error tolerance is set to $\varepsilon = 4\sigma_n$. Additionally, the figure shows the possible gains for the maximum and the minimum transmission power levels (see Section 2.3.1), so as to respectively obtain the upper and lower bounds on the achievable performance (solid lines are used to indicate maximum transmission power, dashed lines to indicate minimum transmission power).
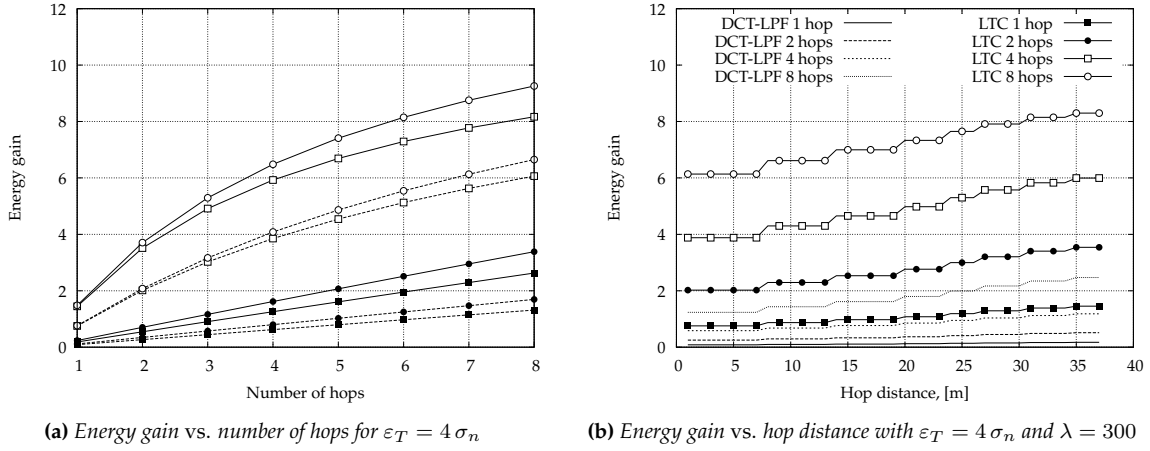
(a) *Energy gain* vs. *number of hops for* $\varepsilon_T = 4\,\sigma_n$

(b) *Energy gain* vs. *hop distance with* $\varepsilon_T = 4\,\sigma_n$ *and* $\lambda = 300$

Figure 4.8. *Energy efficiency. Comparison of selected lossy compression schemes for a multi-hop scenario.*

As might be expected, the energy gain increases with the number of hops. This is because, although the energy spent on the compression at the source node is comparable to that spent on the transmission, the compression energy cost is incurred only at the source node; while each additional relay node has to deal only with compressed data. Note that DCT-LPF is not energy efficient in single-hop scenarios, but it can actually provide some energy gains when the number of hops is large enough (e.g., larger than 2), and the transmission power is set to the maximum level. For the minimum transmission power, DCT-LPF starts being energy efficient only after $5 - 6$ hops.

In fact, DCT-LPF performs better in terms of compression ratio than LTC, which is more efficient in terms of complexity. As the compression energy is a fixed cost and the gain increases progressively with the number of hops, more aggressive compressing implies that at some point the extra processing cost may be compensated with a reduction in the (re)transmission energy, which is associated to a better compression ratio. However, for this particular technology, this balance occurs above 40 hops, an unrealistic value in practical networks.

Fig. 4.8b shows the maximum achievable energy gain versus the distance between hops. Given the distance, the transmission power is selected according to the Friis path loss formula (with path loss exponent $\alpha=3.5$, which is typical for WSNs [46]). This is done by considering the transmission power levels and the receiver sensitivity $P_{th} = -95$ dBm of the CC2420 transceiver [20]. For each value of the distance, the energy gain is evaluated using the minimum transmission level that leads to a received power above P_{th}. As shown in Fig. 4.8b, the energy gain increases with the distance, as the transmission power becomes progressively higher than that needed for compression. This effect becomes more pronounced when the number of hops is increased, as the relay nodes only have to forward the data (no processing), thus benefiting from the smaller number of bits to be received and transmitted.

4.4.5 Correlation, Compression and Accuracy

This section uses numerical fitting to investigate the relationships between the achievable compression ratio η, the relative error tolerance ξ and the computational complexity N_C. Note that until now η has been understood as a performance metric which depends on the chosen error tolerance $\eta = \eta(\varepsilon_T)$. This is equivalent to considering the relative error tolerance ξ as an input parameter for the compression algorithm, since $\varepsilon_T = \xi \sigma_N$. The idea now is to interchange the mathematical relationship between η and ξ by conversely thinking of ξ as a function of η, which is now considered an input parameter, $\xi = \xi(\eta)$.

This mathematical rearrangement has important implications. In energy-constrained devices, the scarce resource is energy. Then, the most immediate consequence is generally that the amount of energy is upper-bounded; i.e., a system has an energy budget for its entire operational life. From this new perspective we can see that once the energy available is fixed and, thus, the required compression ratio as well, the expected reconstruction error can be evaluated as $\varepsilon = \varepsilon(\eta)$. Note that, in this expression, the compression ratio required for a given energy constraint is a function of the energy bound $\eta = \eta(\mathcal{E})$. This essentially means reinterpreting lossy compression as a *Quality-of-Service* (QoS) technique, in which the quality metrics are given by the reconstruction accuracy ε. The constraint, in this case, is driven by the available energy $\mathcal{E}$.

Reconstruction Accuracy

Quantitatively, the relative error tolerance ξ has been related to the compression ratio η for the best compression methods, namely LTC and DCT-LPF, through the following polynomials:

$$\xi_\lambda(\eta) = \begin{cases} \dfrac{p_1\eta^2 + p_2\eta + p_3}{\eta + q_1} & \text{LTC} \\ \dfrac{p_1\eta^4 + p_2\eta^3 + p_3\eta^2 + p_4\eta + p_5}{\eta + q_1} & \text{DCT-LPF}, \end{cases} \tag{4.5}$$

where the fitting parameters $p_1, ..., p_5$, and q_1 are computed for each specific correlation length λ. The relationships were found through numerical fitting, which involved running extensive simulations with synthetic signals. The numerical values of the fitted parameters and details about the process can be found in [J2].

Fitting results have been validated against real world-signals, which were obtained from the environmental monitoring WSN testbed, deployed on the ground floor of the Department of Information Engineering (DEI) at the University of Padova, Italy [21]. This dataset consists of measures of temperature and humidity, sensed over a 6-day period at a sampling interval of 1 minute. Measured correlation lengths are $\lambda^{(T)} = 563$ and $\lambda^{(H)} = 355$ for temperature and humidity signals, respectively.

The empirical relationships of Eq. (4.5) are shown in Fig. 4.9a and 4.9b through solid and dashed lines, whereas the markers indicate the performance obtained by applying LTC and DCT-LPF to the
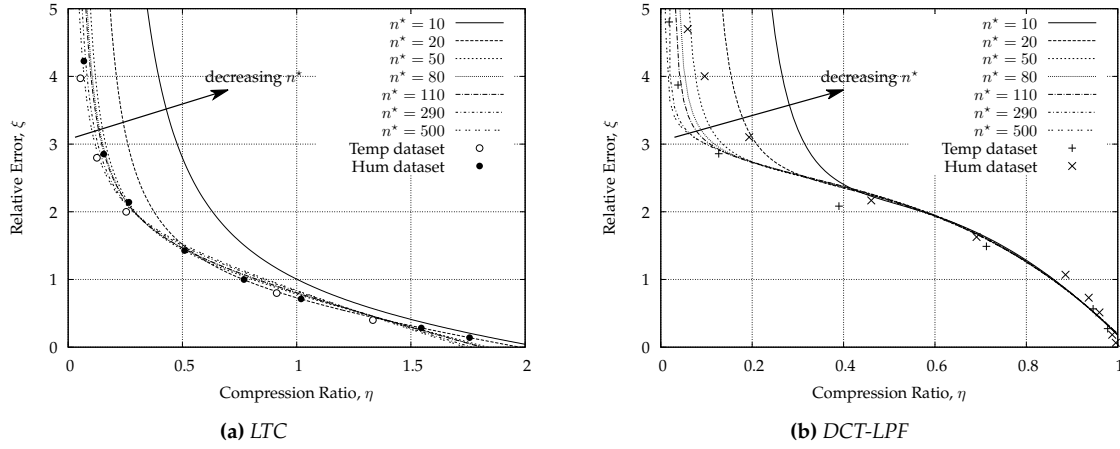
Figure 4.9. *Fitting functions $\xi(n^\star, \eta)$ vs. experimental results.*

considered real datasets. As can be noted from these plots, although the numerical fitting was obtained for synthetic signals, Eq. (4.5) closely represents the actual tradeoffs. Moreover, this implicitly provides additional verification of the synthetic generation method.

Regarding the relations found, decreasing λ seems to have a low effect, as the curves relating ξ to η remain nearly unchanged in terms of functional shape and are merely shifted toward the right. Finally, the dependence on λ is particularly pronounced at small values of λ, whereas the curves tend to converge for increasing correlation lengths (larger than 110 in the figure).

Computational Cost

Regarding the computational energy cost, observations indicated that the number of CPU operations for compressing the time series scales linearly with η for both LTC and DCT-LPF (See Fig. 4.5b and Fig. 4.6b). Hence, assuming that the number of cycles per compressed bit $N_{\mathcal{I}}^{(b)}$ exhibits a linear dependence on both λ and η, it can be expressed through a polynomial as follows (the fitting coefficients are shown in Table 4.3):

$$N_{\mathcal{I}}^{(b)}(\lambda, \eta) = \alpha\eta + \gamma\lambda + \beta \ .$$

Note that the dependence on λ is much weaker than that on η. Thus, for practical purposes, it can be neglected without any important loss of accuracy. In fact, DCT-LPF appears to show a one-to-one mapping (in a statistical sense) between any target compression ratio and the number of DCT coefficients that are required to achieve this target performance. The computational complexity is directly related to this number of coefficients.

Method	Fitting coefficients		
	α	β	γ
LTC	16.1	105.4	$3.1 \cdot 10^{-16}$
DCT-LPF	$48.1 \cdot 10^3$	82.3	$-2 \cdot 10^{-13}$

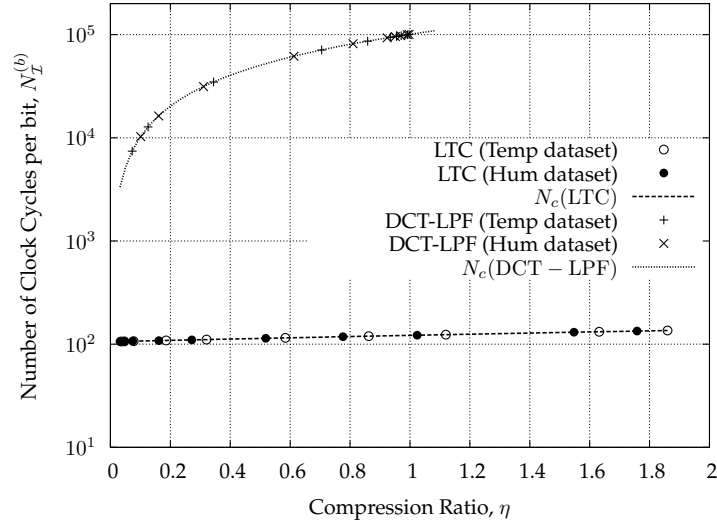**Table 4.3.** *Fitting coefficients for $N_{\mathcal{I}}^{(b)}(\lambda, \eta)$.*

Figure 4.10. *Fitting functions $N_{\mathcal{I}}^{(b)}(\eta)$ vs. experimental results for LTC and LPF-DCT.*

For LTC, the dominating term in the total number of operations performed is η, as this term is directly related to the number of segments that are to be processed. With these considerations, the remainder of this section uses the simplified relationship:

$$N_{\mathcal{I}}^{(b)}(\eta) = \alpha\eta + \beta \ . \tag{4.6}$$

The accuracy of Eq. (4.6) is verified in Fig. 4.10, which shows empirical approximations against the results obtained for the real world signals described above.

Technology Trade-offs

The above empirical expressions are used to generalize the results to any processing and transmission technology by separating out technology dependent and algorithm-dependent terms. Eq. (4.7) is obtained by introducing the numerical fitting Eq. (4.6) in Eq. (4.2). Constants α and β are the algorithmic dependent fitting parameters indicated in Table 4.3. So this inequality is notably only a function of η, once the compression algorithm has been selected.

$$\frac{E_T^b}{E_{\mathcal{I}}} > \frac{N_{\mathcal{I}}^{(b)}(\eta)}{1-\eta} = \frac{\alpha\eta + \beta}{1-\eta} \; , \tag{4.7}$$

Eq. (4.7) can be used to assess whether a compression scheme is suitable for a specific device's architecture. As an example, Fig. 4.11 shows the impact on compression efficiency of different processor technologies and radio settings. The TI MSP430, in particular, with a featured consumption of $380\mu A/Mhz$ ($E_{\mathcal{I}}$=1.25 nJ at 3.3V), is compared with a Wonder Gecko EFM32WK Cortex-M4, rating a consumption of $230\mu A/Mhz$ ($E_{\mathcal{I}}$=0.76 nJ at 3.3V). Regarding the radio, the selected CC2420 transceiver has been configured at full speed $250kbps$ (E_T^b=230 nJ at 3.3V), and low speed $20kbps$ (E_T^b=2875 nJ at 3.3V).

Fig. 4.11a shows the results for the reference platform used in this chapter (MSP430 processor and radio 250kbps). The technological energy ratio for this platform is $E_T^b/E_{\mathcal{I}}{\simeq}210$. The numerical evaluation of Eq. (4.7) for DCT-LPF reveals that this compression scheme is inefficient for any value of η, i.e., the overall energy expenditure due to transmission plus compression is higher than the energy spent in cases where compression is not applied. (In Fig. 4.11a, DCT-LPF does not enter in the *efficient region* for any value of λ). Instead, LTC provides energy savings for $\eta \leq 0.45$ (See also Fig. 4.7a).

These results can be generalized to any other device technology, by comparing Eq. (4.7) against the corresponding ratio $E_T^b/E_{\mathcal{I}}$ and checking whether the inequality holds. The same radio configuration is used in Fig. 4.11b, but compression is computed in a more efficient processor. As the radio is unchanged, the raw transmission cost is the same (vertical dashed line). However, using a processor that demands less power enables some additional algorithms to be efficient as well.

The scenario is completely different in Fig. 4.11c and Fig. 4.11d, where a lower bit rate is used, meaning that transmission time is much longer. In this case, the technological ratio between processing and transmission is one order of magnitude better, and it makes local processing much more attractive by allowing several new algorithms to enter in the efficiency region.

Reconstruction Accuracy Trade-Offs

On the basis of the above results, the reconstruction accuracy for a given energy bound can be estimated as follows. From Section 4.1.2, the energy for compression is given by:

$$E_C = N_b(x)N_{\mathcal{I}}^{(b)}(\eta)E_I,$$

where the number of instructions per uncompressed bit (which is in fact characteristic of each specific algorithm) has now been expressed as a function of the compression ratio $N_{\mathcal{I}}^{(b)}{=}N_{\mathcal{I}}^{(b)}(\eta)$. Regarding the transmission energy, it can be expressed in terms of the compression ratio as:

$$E_T = N_b(x)\,\eta\,E_T^{(b)},$$

(a) *MSP430, 250kbps, $E_T^b/E_\mathcal{I}$=210*

(b) *CortexM4, 250kbps $E_T^b/E_\mathcal{I}$=346*

(c) *MSP430, 20kbps, $E_T^b/E_\mathcal{I}$=2633*

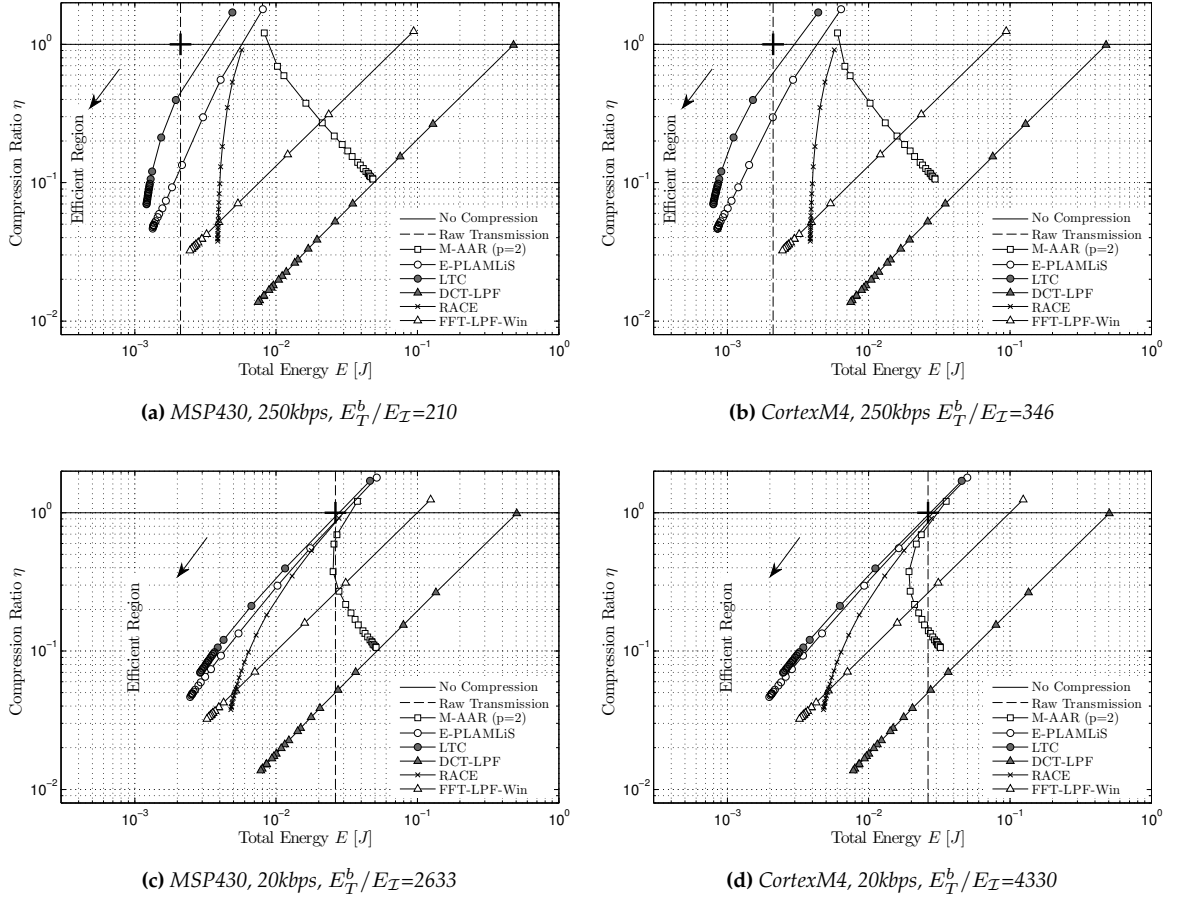**(d)** *CortexM4, 20kbps, $E_T^b/E_\mathcal{I}$=4330*

Figure 4.11. *Energy efficiency. Comparison of lossy compression schemes for different technologies.*

The total energy is obtained from the contribution of both terms, as expressed in Eq. (4.8)

$$E = N_b(x) \cdot \left(E_I N_\mathcal{I}^{(b)}(\eta) + E_T^{(b)} \eta \right) = f(\eta) \tag{4.8}$$

Eq. (4.8) expresses the overall energy as an explicit function of the compression ratio $E=f(\eta)$. This equation can be inverted using the value of $N_\mathcal{I}^{(b)}(\eta)$ from Eq. (4.6), which leads to the desired relation $\eta=\eta(E)$. Once the specific value of the required compression ratio is obtained from the energy bound $\hat{\eta} = f^{-1}(\hat{E})$, the expected reconstruction error can finally be obtained using the function $\xi(\lambda, \eta)$.

$$\epsilon_T(\lambda) = \xi(\hat{\eta}, \lambda) \cdot \sigma_N \tag{4.9}$$

To evaluate Eq. (4.9), the numerical value of $\xi_\lambda(\hat{\eta})$ is obtained from Eq. (4.5). Note that the correlation length λ must be fixed for this last evaluation. Furthermore, the effect of λ can be significant for low values of λ (see Fig. 4.10).

Fig. 4.12 shows an example of how this methodology can be applied. In this example, the system continuously acquires records of 500 samples at a rate of N_{RCD} records per second. The question
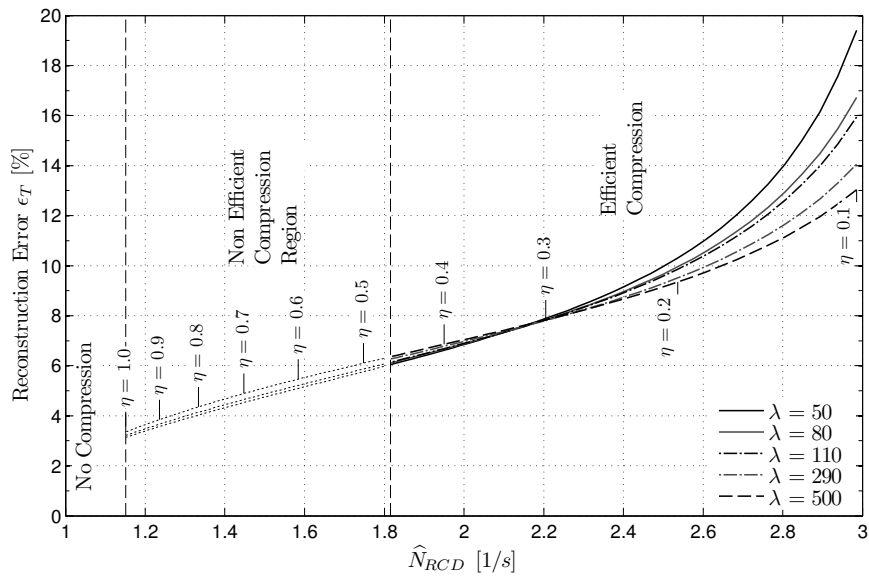
Figure 4.12. *Reconstruction accuracy, measured as a function of the maximum achievable sampling rate.*

arises about what maximum sampling rate is allowed while still meeting specifications. For this example, the expected life for the system has been fixed to 1 year, and the system is equipped with a $10Ah$ battery.

The figure shows three regions. The first one is to the left of the vertical dashed line, where the system can operate while continuously acquiring and transmitting data, without any need for compression. The second region shows the interval in which compression is possible but is however not effective. This means that raw transmission of all samples is still feasible and, subsequently, reconstruction would be perfect. The third region, above $\eta = 0.46$, shows application settings in which raw transmission would not be viable. Above N_{RCD}=1.8 records per second, the system requires some compression in order to be sustainable. Fig. 4.12 shows the reconstruction error to be expected when the required compression is applied under such conditions.

Discussion

This chapter presents an in depth analysis of the potential advantages offered by lossy compression algorithms for time-varying signals. It does so by considering recently proposed and lightweight schemes such as Lightweight Temporal Compression (LTC), as well as the more sophisticated FFT- or DCT-based techniques. This analysis examines the relationships between computational complexity, overall energy consumption and signal representation accuracy.

In a scenario where reconstruction fidelity and signal statistics play a fundamental role, a methodology is designed to assess whether signal compression actually helps in the reduction of the overall energy consumption, which itself depends on the compression algorithm, the chosen reconstruction fidelity, the signal statistics and the hardware characteristics.

The results reveal that compression schemes can be inefficient in terms of energy consumption when they account for the energy required by the compression computation. Instead, tiny linear methods such as LTC lead to substantial savings in terms of energy expenditure, while at the same time leading to satisfactory compression ratios, reduced network delay and increased reliability performance.

Previous considerations have taught us that signal compression may in fact provide some energy savings. However, its usage should be carefully evaluated. Processing and transmission costs can be of the same order of magnitude and, depending on the specific algorithm, the former may even dominate the latter. Furthermore, the answer to the question of energy efficiency is closely related to the current technology in use which may change from platform to platform. Even more disturbing, the answer is related not only to the particular algorithm and platform used, but to the characteristics of the specific signal being compressed.

In the last section of this chapter, lossy compression is analyzed from an alternative point of view. In this new approach, compression ratio is not seen as a performance metric, but as a control parameter for adjusting the power consumption of the system. Even more, with the aid of numerical fitting methods, the reconstruction error is estimated from the compression ratio. Therefore, this method makes it possible to predict reconstruction fidelity as a function of energy consumption.

Chapter 5

Lean Sensing

Event-driven applications are used to monitor the occurrence of certain events. In general, events are inherently stochastic, and the main function of the system is precisely to detect and report the occurrence of such events. Strategies like data compression (in which events are essentially binary entities) or local processing (where detected events must be reported) are not suitable for these kinds of applications.

But, what does randomness mean? Let us focus on one familiar example. The occupancy of a parking spot at a given instant is unpredictable, as are human beings. However, when the state of a set of related spots is aggregated, a correlation structure emerges. What is more, past occurrences reveal a high degree of auto-similarity (as with human behavior) when studied within the proper temporal scale.

Throughout this chapter, we propose introducing contextual information in order to optimize power consumption in those applications to which classical policies are not applicable. The discussed techniques include customizing the sample to temporal correlations (i.e, to adapt sensor behavior to the expected activity) and inferring the system-state based on spatial correlations (i.e., reconstructing the state of some specific locations from the information already gathered on the surroundings when their data is not available).

Thus, the aim of this chapter is to evaluate the impact that a number of system-level optimization strategies have on power consumption. Furthermore, the ultimate objective is to evaluate for these methods the trade-off between power consumption and the accuracy of the monitoring service provided, for both single devices and overall system performance.

5.1 Introduction

THIS chapter will use the smart parking application described in Section 2.5.2 to introduce alternative optimizations methods based on the contextual information of the system. In this chapter, *system* is understood to be a set of spatially-related sensors working together to provide a monitoring tool (based mainly on temporal aggregated information). The main objective of these techniques is precisely to harness the (co-)relations found within the data.

For this purpose, it is first necessary to introduce the particular characteristics of this application and how the monitoring service is provided, in order to extract the real requirements of the application from the user's point of view, i.e., the usage experience. This perspective can be different from the technical view of an engineer. Second, the intrinsic structure of parking patterns is described. Finally, the chapter concludes with an analysis of some energy optimization strategies drawn from the learned structure. These strategies are evaluated on the basis of system-level metrics, i.e., metrics defined to quantify the quality of the monitoring service.

This chapter will use real-world datasets from two different cities, each having several hundred sensors deployed. These two actual case-studies are not only application examples, but they settled the foundations for the pursuit of the presented methodology. The development starts with the adoption of a system-level perspective, i.e, a perspective of the monitoring service that is designed to identify and visualize application patterns. Then, this perspective should help us better understand the application itself, with the confidence that the patterns found can be used to optimize the energy efficiency.

5.1.1 Scenario

Most of us have suffered the experience of driving into a town looking for somewhere to park. This is a very familiar problem that citizens around the world must bear every day. As a result, we have developed some (sometimes rather efficient) strategies for dealing with it. These strategies are usually based on previous experience and probably on some intuitions we may have into an underlying structure.

In theory, the state of an individual parking bay can be considered a stochastic process [6]; thus, the occupancy of an individual parking bay is basically random. However, we intuitively understand that parking availability depends strongly on the quantity and diversity of spaces, time of day, number of other drivers, specific area regulations and many other undetermined variables which yield to specific characteristic behaviors. For instance, on a city-level scale and using a central area as an example, parking demand tends to increase early in the day, hold at a relatively high level through-

out the mid-day, and then spike in the evenings when drivers pursue leisure activities (restaurants, cinemas, theaters, etc.).

On-street parking sensors have been utilized in different cities around the world to improve the efficiency in the use of public parking. Representative examples are San Francisco, Los Angeles, Moscow, Nice, London, and Barcelona, among many others, where small sensing devices have been deployed in every parking bay within large monitored areas.

The benefits of these devices are undeniable; they have proven to be very useful in helping cities better understand parking patterns, and they have become a very valuable tool in optimizing parking operations as well [22]. Moreover, they help drivers find parking spaces more efficiently by reducing the hustle and bustle of traffic, and they provide better user experiences.

However there is a common concern about the cost-benefit relationship of this solution in many cities. The deployment of thousands of sensors involves a large upfront investment by public administrations or parking operators, and it imposes additional complexity on the operational side. As a result, some previous studies have analyzed the cost-benefit relationship of such a solution in cities like Los Angeles or Indianapolis [22].

Some other studies take a more conceptual approach and have built a model to reduce deployment costs by providing sensor readings on only a fraction of the parking spots in an area, followed by the use of extrapolation to calculate city-wide saturation levels [32].

But none of them have centered their efforts on adapting the technology to ensure optimal operations of the whole network for a given return of investment period. This entails a threefront strategy: first, optimizing the production cost associated with battery size; second, extending the operational life by means of more efficient technologies; and third, minimizing unplanned interventions.

5.1.2 System-Level Performance

Great efforts have been made to reduce consumption of every device by compressing the information to be distributed or even by not distributing the information at all, but making the right decisions that are locally based on the information available. Yet sometimes this is inherently not possible. In event-driven applications, there is no reasonable alternative to transmitting every single event. On top of that, many events are often barely reported, so compression techniques are useless.

Consider the example of parking sensors. Each device periodically wakes-up to check the state of the spot. When a car is detected, the sensor communicates the event to the data-collection center so that system managers can keep track of the state in real time. The minimum amount of information is encapsulated in each message, as the state can be codified with a single bit. Therefore, it is intrinsically incompressible. Furthermore, the event should be reported, as the main function of these devices is precisely to monitor the occurrence of these events; so it cannot be considered an

alarm-based application such as those discussed in Chapter 3, because mobility agents may require some kind of real-time information. Under these circumstances, it seems difficult to find alternatives for saving energy other than using more efficient technology.

However, little attention has been paid to the application itself and to how the monitoring service is provided. Monitoring systems are typically made up of two main layers [77]. The hard infrastructure is composed of thousands of networked embedded devices, whose operation (although interconnected) is in essence independent. Above this hard layer, software and service technologies enable high-level access and utilization of the real-world data and resources. This is the layer to which we pay attention here.

In this specific application, it turns out that individual data from a sensor is not really relevant. Instead, the key information needed is the aggregate of certain regions (streets, blocks, etc. what we call sectors) and the time-averaged information of these sectors (e.g., hourly occupancy of a street, etc.). This characteristic will drive us to a set of possibilities that may in fact reduce power consumption while maintaining acceptable accuracy of the information provided.

5.1.3   Performance Indicators

Choosing the right performance indicator is essential for the evaluation of any application, system or activity while at the same time relying upon a clear understanding of the business model [J1]. In this case, these performance metrics must be found within the system user's experience, not from the viewpoint of the technology provider. Based on WorldSensing's experience as a provider of Smart Traffic Management Solutions, several possibilities exist, of which the following can be considered the most representative indicators:

$\mathcal{D}$ **Occupancy:** System occupancy is not related to the specific state of a particular parking spot, but to the percentage of time a spot is occupied over a period of time. For traffic management, the historical occupancy of the spot provides more information than the current state at a specific time. Likewise, the mean occupancy of an area provides more information than the state of a single sensor. Occupancy is the most intuitive and useful indicator for this particular application.

$\mathcal{D}$ **Turn-over:** Is a direct indicator of the activity in a specific area. A high activity area fosters the absorption of traffic, while a quiet area can be problematic, especially if the occupancy is high. Again, activity makes sense over a period of time, not as single events.

One important issue that can be considered transversal to the previous definitions is the idea of *repeatability*. Without pattern recurrence, predictive management is not possible. So in the end, the degree of repeatability sets the limit between predictive management (making decisions before things occur) and reactive management (making decisions based on what is actually happening).

5.1.4 Energy Model

This application is a particular case of the simple reporting application described in Section 2.5.2. In terms of power consumption, the application is determined mainly by two contributions: periodic sampling and the reporting of physical events. These tasks, based on the repetitive operation, are parametrized as follows:

- A record of N_S samples is acquired with a fixed interval time, T_{RCD}.

- Events occur with a characteristic elapsed time, T_{MSG}.

As explained in Section 2.3.1, when sensors generate endogenous traffic according to some stochastic process distribution, the time elapsed between consecutive messages, T_{MSG}, should be characterized by an appropriate statistical estimator (often a simple average is enough). This approximation is reasonably good for long-term averaging.
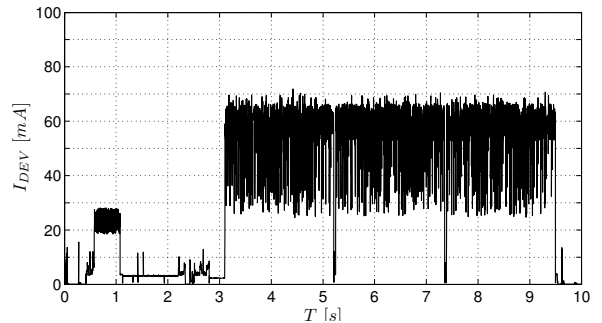
Sampling Characterization

For each record composed of N_S samples, the total time the sensor remains switched on is given by $T_{SNS}^{(ON)} = T_S \cdot N_S$, where T_S is the physical sampling interval (See Fig. 5.1a). In many applications, the sampling rate is fixed by filter requirements.

The current of the acquisition block can be obtained by averaging the charge to get the N_S samples of the record over the time elapsed between consecutive records T_{RCD}, i.e. the wake-up period. In Eq. (5.1) $\bar{Q}_{SNR}$ is the average charge to get one sample (see Section 2.4.2) and comprise both the sensor and the ADC conversion. $\bar{I}_{SNS}^{(STB)}$ accounts for the stand-by or quiescent current of the sensor.

$$\bar{I}_{ACQ} \cong \frac{\bar{Q}_{SNR} \cdot N_S}{T_{RCD}} + \bar{I}_{SNS}^{(STB)} \tag{5.1}$$
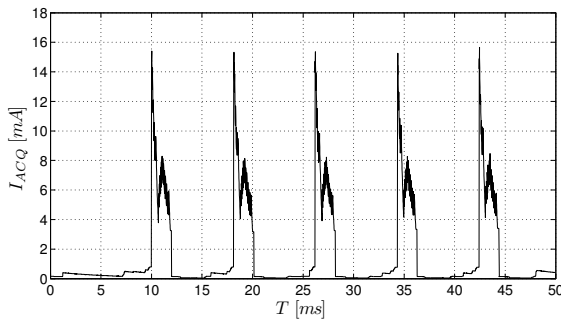
(a) *Record composed of N_S=5 samples elapsed T_S=8 ms*

(b) *Radio message: initialization and 3 retransmissions.*

Figure 5.1. *Contributions for the energy model characterization.*

Point to Point Communications

The average power of the communications block can be expressed in terms of the charge required to send a radio message and a characteristic time between consecutive events T_{MSG}. The reported information is aggregated into a unique message. We assume that this message is retransmitted a certain number of times (N_{RTX}) to increase the probability of success (See Fig. 5.1b).

$$\bar{\mathcal{I}}_{NET} \approx \frac{N_{RTX} \cdot \bar{\mathcal{Q}}_{MSG}}{\bar{E}\left[T_{MSG}\right]} \doteq \frac{N_{RTX} \cdot \bar{\mathcal{Q}}_{MSG}}{\widehat{T}_{MSG}} \tag{5.2}$$

Joint Model

Eq. (5.3) combines the two main contributions of these applications: sensing and communications. As this is essentially a bare reporting application, the cost associated with processing can be considered negligible. Recalling the meaning of each individual contribution from the fitting in 2.4, α can be interpreted as the charge per sample $\bar{Q}_S$; γ is an estimator of the average charge per message $\bar{Q}_{MSG}$, while δ basically accounts for OS management consumption and quiescent DC/DC currents.

$$\bar{I}_{DEV} = \frac{\alpha N_S}{T_{RCD}} + \frac{\gamma N_{RTX}}{\widehat{T}_{MSG}} + \delta \tag{5.3}$$

5.1.5   Operational Cost and Energy Constraints

The classic limitations of industrial wireless monitoring applications have been deeply analyzed in the literature. They include the power consumption of wireless sensors, the ease of handling communication interfaces and global coverage, among others. However, including operational costs in the design flow of IoT devices is still scantily practiced among researchers, although these costs are especially relevant in critical infrastructures. This has been identified as a key issue in developing the IoT. Therefore, we will make some remarks here, from an industrial point of view.

Common energy optimization approaches use the same policies independently of the observed level of activity in a specific area. Even though the battery life of an individual device may be extended by using the described approaches, some devices will still deplete their power more quickly than others. Differences in battery usage across the same network become troublesome if we consider that battery replacement interventions must be scheduled. In that regard, it must be taken into account that each intervention costs money and time. In some scenarios (such as smart cities), it may actually become an annoyance to citizens. Of course, it may be possible to simply replace all batteries regardless of their power level at each intervention, but that would be unnecessarily wasteful in terms of both cost and environmental impact. Therefore, we believe that it is very important to not

only increase battery life, but to also homogenize battery consumption across all devices in order to facilitate interventions and reduce maintenance costs [C1].

Given the above reasons, this work considers the expected battery life to be a constraint (not a variable to be maximized), and that the objective of the energy policy should be to ensure that the minimum expected life-time specification is met. With this aim, Eq. (5.3) can be interpreted as a parametric function binding together T_{RCD} and T_{MSG}. More specifically, when the expected battery life is constrained and the conditioned maximum I_{DEV} is thus fixed, this equation defines one-to-one mapping of the expected activity to the maximum allowed sampling rate.

$$T_{RCD}^{(MAX)} = f(\widehat{T}_{MSG}) \tag{5.4}$$

Fig. 5.2 shows a numerical simulation of Eq. (5.3). The black lines represent the mapping f in Eq. (5.4), once I_{DEV} has been set so as to achieve the desired uninterrupted operation (T_L).
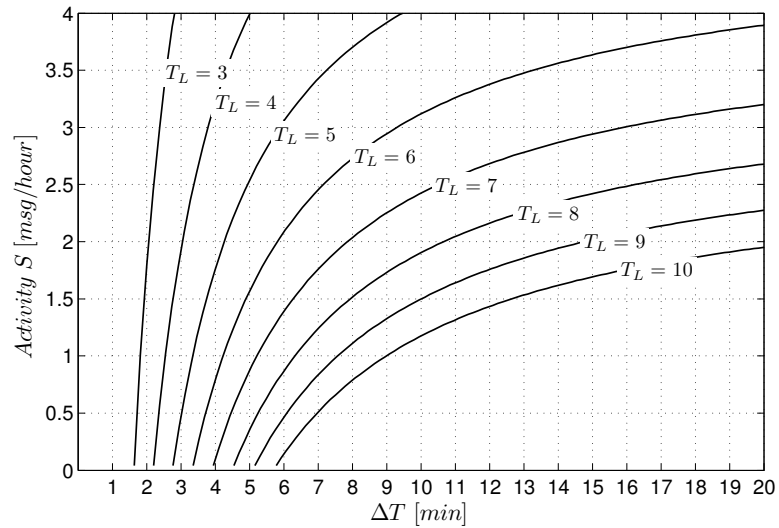

Figure 5.2. $T_{MSG} \leftrightarrow T_{REC}$ *mapping for different values of expected life-time T_L (in years).*

5.2   Monitoring Service and Data Structure

In this chapter we address *Lean Sensing*, which aims to improve the cost-benefit from a system wide perspective taking into account the predictable behavior of the agents involved in the target application. The first part of this chapter presented the modeling framework for individual agents. This section aims to provide a global vision of a system composed of hundreds or even thousands of devices actuating autonomously but sharing a common goal.

The system here described is designed to identify and visualize patterns in a complex, macro-scale monitored process. Notably, as next section shows, it is also in these patterns where one can find some keys to support design of efficient energy management strategies.

Throughout this chapter, it is assumed that the monitoring service provided should drive optimization policies. It is then necessary to obtain a clear understanding of how these management tools operate. An analysis of the smart city ecosystem reveals some generally desirable features [59]:

- Planning is performed not on snapshots but on accurate historical data from dynamic resources with measurable effects.

- Patterns can be identified, and the real reasons why and what is happening can be identified, leading to better knowledge of the system.

- Cities can monitor and react in real-time to citizen needs, events and incidents.

It is thus understood that, although the system administration maintains real-time monitoring as a feature, it is based on a comprehensible representation of historical data. The metrics defined here and the developed visualization tools are both oriented towards this strategy.

5.2.1   System State: Occupancy

As stated before, occupancy is related to the percentage of time a spot stays occupied. However, the instantaneous state of a specific spot is not particularly useful. The city is a complex, dynamical system, and the state of a specific spot can randomly change at any instant. Instead, sector-aggregate information can be much more valuable for both monitoring as well as planning purposes.

To deal with these concepts more formally, the following definitions will be utilized in the remainder of this chapter:

$\mathscr{D}$. **Sensor Occupancy:** S is the time ratio a specific spot has been occupied during a time-slot. It is defined as a multi-dimensional variable, $S_{i,j}^{(k)} \in [0,1]$; where index $i=1...N_H$ accounts for the time-slot during the day (typically the slot is one hour); $j=1...N_D$ denotes the day of the monitored period; and finally, index k refers to the specific sensor.
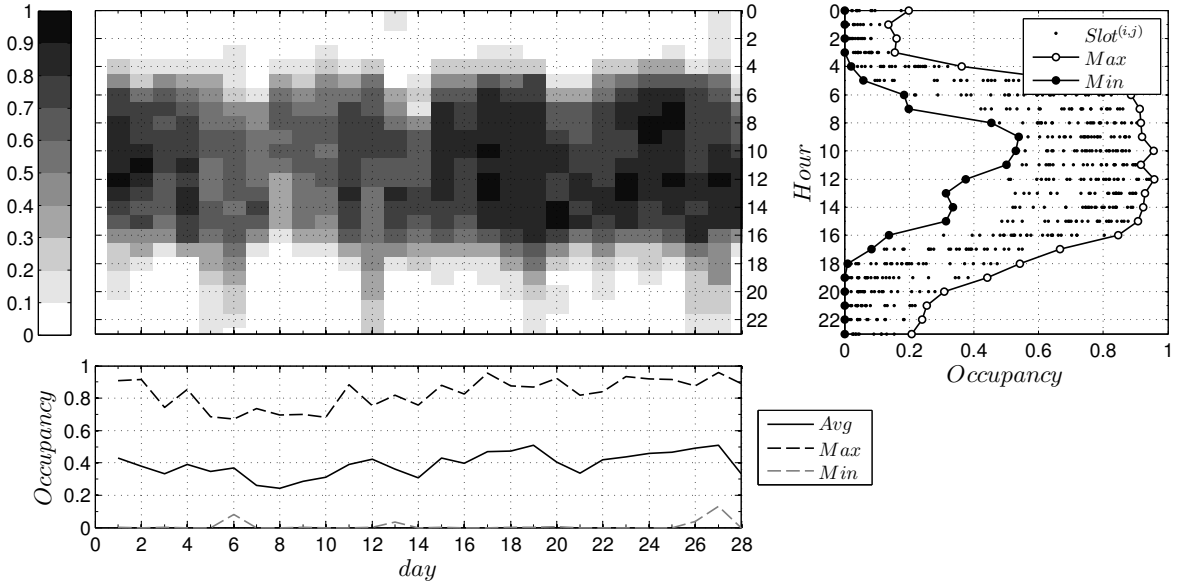
Figure 5.3. *Monthly average occupancy in a metropolitan sector, from different perspectives. Time-slots of 1 hour.*

$\mathscr{D}$. **System Occupancy:** the system occupancy state is obtained by averaging considered sensors in each time slot, N_S being the total number of sensors that are aggregated. This concept can be applied to the whole system, to a sector or to any subset of sensors.

$$S_{i,j} = \frac{1}{N_S} \sum_{k=1}^{N_S} S_{i,j}^{(k)} \tag{5.5}$$

$\mathscr{D}$. **Occupancy Reconstruction Error:** the difference between the system occupancy state, which is built upon a complete knowledge of the system, and the state inferred with partial information. Specifically, the reconstruction error is defined in Eq. (5.6), $\widehat{S}$ being the system occupancy state estimated from incomplete information.

$$\epsilon_S = \frac{1}{N_H \cdot N_D} \sum_{i=1}^{N_H} \sum_{j=1}^{N_D} |S_{i,j} - \widehat{S}_{i,j}| \tag{5.6}$$

Fig. 5.3 shows the system occupancy state registered in a metropolitan area during one month (February). The colormap represents the average occupancy of nearly 500 monitored spots. In this figure, $\bar{S}$ is displayed hourly in the vertical direction and daily in the horizontal.

To make the historical evolution more understandable, data is projected in two orthogonal directions. First, hourly occupancy has been averaged over each day in the figure below. Then, this plot represents the daily mean evolution of the full period. Additionally, maximum and minimum daily occupancy have been represented. The second one, on the right-hand side of the scale map, is a projection of the hourly occupancy of every single month in the period. This figure captures with

a single snapshot the historical hourly behavior, showing the variability of daily patterns during the period being studied.

Taking this view can help to understand several patterns of citizen behavior: occupancy is much higher during daytime hours and is almost zero at night (sensors are deployed in a commercial area); weekend specific patterns are noticeable, and they are appreciably different from the rest of the week; finally, lower occupancy was registered in the first week of the period, most likely due to adverse weather conditions. These are some examples of the kind of information typically required for intelligent traffic management.

5.2.2   System Activity and Turn-Overs

The activity (i.e., the number of events that happened in a given slot of time) and the number of turn-overs (which is directly related to the former) are essential indicators for mobility agents. Intuitively, activity provides an estimate of the flux of traffic that can be adsorbed. For instance, in parking guidance systems, drivers can be redirected to areas whose activity is expected to be higher. Even more, activity can be used for adjusting prices and bounding minimum and maxim parking-times in paid parking areas.

Due to its relevance, activity should be well defined and precisely quantified. So this chapter formally complies with the following definitions:

$\mathscr{D}$. **Sensor Activity:** the sensor activity R is the number of events occurring in one spot during a specific time-slot. $R_{i,j}^{(k)}$ is defined in a similar way as the occupancy. The index $i=1...N_H$ designates the time-slot, $j=1...N_D$ indicates the day, and k is the sensor index (See Fig. 5.4).

$\mathscr{D}$. **System Activity:** the average activity of a set of sensors in each time-slot:

$$R_{i,j} = \frac{1}{N_S} \sum_{k=1}^{N_S} R_{i,j}^{(k)} \tag{5.7}$$

$\mathscr{D}$. **Activity Reconstruction Error**: the error made while estimating the system activity with incomplete information $\widehat{R}$.

$$\epsilon_R = \frac{1}{N_H \cdot N_D} \sum_{i=1}^{N_H} \sum_{j=1}^{N_D} |R_{i,j} - \widehat{R}_{i,j}| \tag{5.8}$$

Fig. 5.4 shows the activity recorded in the commercial area of a small town. The displayed subset belongs to the first 28 days of December. The structure of the data is made clear from the picture. The daily pattern indicates higher activity during the morning and afternoon, yet it is very low at night. In addition, week and weekend days can be clearly differentiated. The historical data shows that, in general, activity increases throughout the week as the weekend approaches. However, special behavior is observed on holidays (around December, 25th).
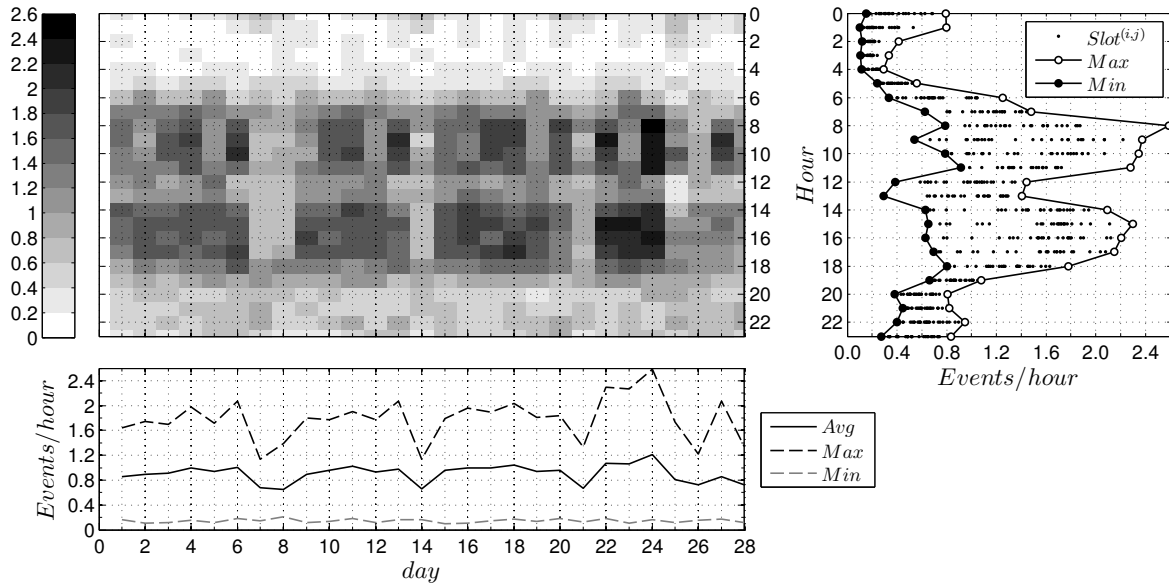
Figure 5.4. *Monthly average activity in a small city, from different views.*

In this application, for which all events detected are directly reported, In this application (for which all events detected are directly reported), the activity is not only related to physical detection, but to the number of radio messages generated. Thus, besides the fact that activity is an important system performance indicator, it has a direct impact on power consumption.

5.2.3   Temporal Stability

In light of the above figures, the question then arises as to how the knowledge of this data structure could be used in some efficient way. The next section exploits some possible applications. However, before doing so, it is necessary to address the issue of repeatability. Repeatability is two-fold. On the one hand, it is essential for predictive management, as it enables making decisions based on past behavior. On the other hand, it provides the key to optimization policies: any engineering decision devised on the basis of patterns observed in past data should still be valid when projected onto the future.

Fig. 5.5 can help clear up doubts about this fundamental question. This figure shows sensor activity averaged over a long period, with the particularity that sensors have been sorted by their activity. This arrangement reveals three different regions corresponding to sensors with low, medium and high activity, and which are separated by two phase transitions. Most importantly, the behavior of three successive months has been superimposed (depicted by three different tones of gray). The self-replicated trend demonstrates an intuitive fact: when a spot has been very active in the past, it remains active. Spot activity is highly dependent on the city environment (commercial parking,

ATM, passing area, etc.). However, in this sense the environment is basically static. Thus, the activity of each spot is expected to be stationary.

This important result ensures the stability of activity as a classification criteria, and thus guarantees that any design workaround that is tailored to a specific class of spots will be valid for the future, provided that the classification criteria is based on activity.

Fig. 5.5 reveals another suggestive pattern. Subplots 5.5a and 5.5b are recorded in two very different environments (a metropolis and a small town). Yet, the distribution is notably almost identical. This result suggests that there are some fundamental patterns underlying the system, independently of the city, and that developed policies may be valid for (quite) different deployments.
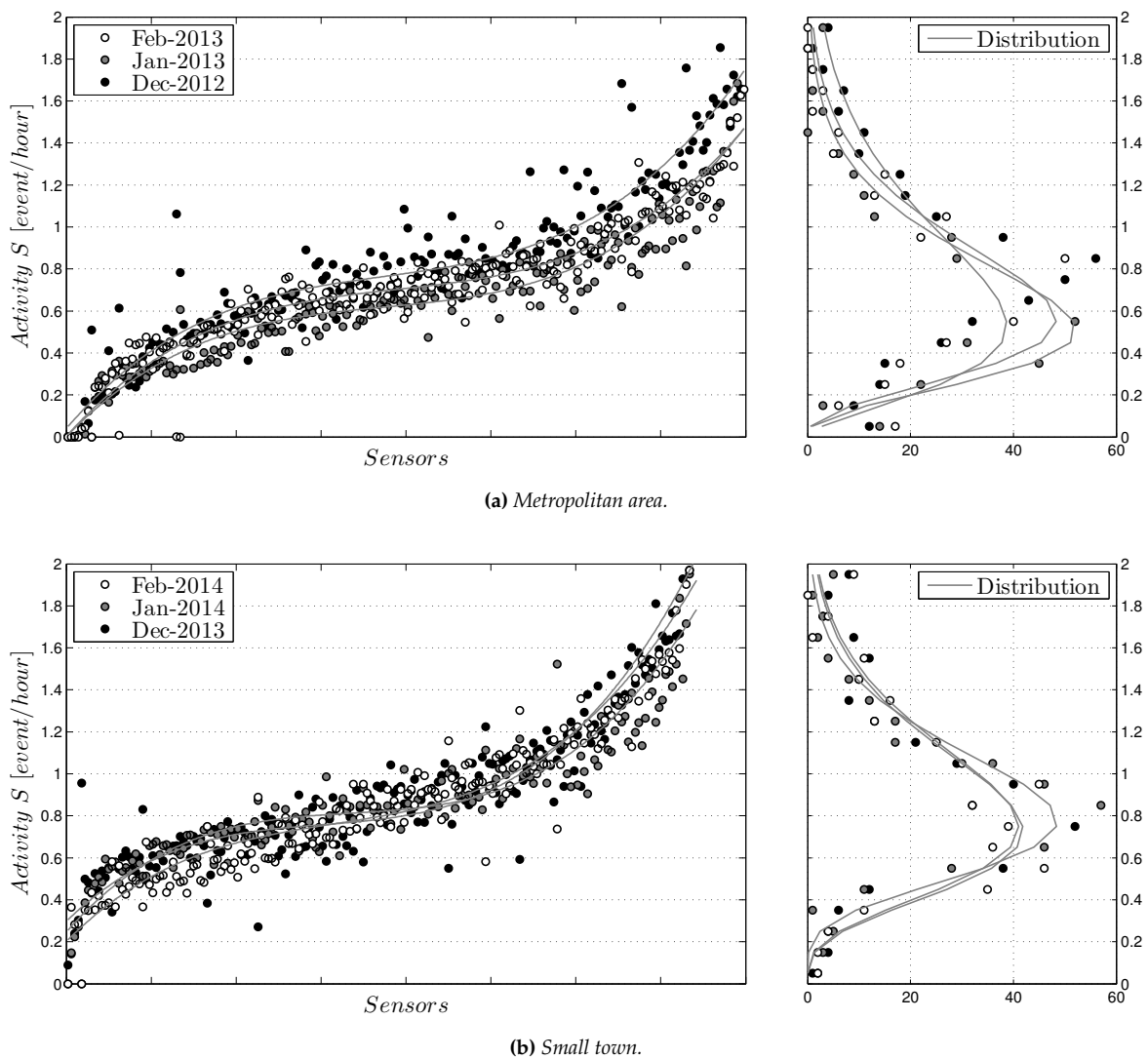

(a) *Metropolitan area.*

(b) *Small town.*

Figure 5.5. *Time evolution of the activity, with individual sensors sorted by activity and their histogram.*

5.3 Performance Analysis of Energy Management Policies

Once analyzed the particular characteristics of this application, i.e., how the monitoring service is provided and the intrinsic structure of parking patterns, this section concludes with an analysis of several optimization strategies. These strategies are evaluated on the basis of the energy model described in Section 5.1.4, with the main focus of quantifying their impact on the service provided under system-level metrics.

5.3.1 Experimental Setup

The platform selected for this study was a variant of the use case described in Section 2.5.2. The main components were the Cortex-M4 32 bit processor, running with an RTOS systick interrupt of 1 ms, and the Honeway HMC5883 magnetometer characterized in Section 2.4.2. This platform was equipped with the Telecom Designs TD1202 long-range radio module.

For the energy model described in Section 5.1.4, the following settings were used: the device took $T_{SNS}^{(ON)}$=60s to determine the state of the spot, requiring a record of N_S=7500 samples acquired at $T_S = 8ms$; Lithium batteries had a total capacity of $30Ah$, with a self-discharge ratio of 1%; finally, the radio module used a non-secure protocol with $N_X = 3$ (re)transmission attempts.

The dataset for this chapter was provided by WorldSensing. Data were gathered over three months from about 1000 outdoor devices, deployed in two different scenarios. Unless stated otherwise, the first month is used for developing, whereas the last two months are used for validation.

5.3.2 Temporal Decimation: Fixed Recording Interval

Classic sampling methods may not be the best choice for systems in which not all sensors experience the same activity. On the one hand, if a fast sampling rate is selected, most of the sensors will deplete their batteries before the scheduled intervention time. On the other hand, if a slow sampling rate is selected in order to assure that all sensors meet the specifications, the monitoring may be too inaccurate. Then, the first issue to address is to quantify the accuracy as a function of the sampling rate, in terms of a suitable metric.

Fig. 5.6 shows the relation between the occupancy reconstruction error $\epsilon_R^{(k)}$ of each single sensor and the activity $S^{(k)}$ (measured as the average interval time between events, $\bar{T}_{MSG}$, for each sensor). The figure shows three sets of points, each corresponding to a fixed interval time of $T_{RCD} = \{5, 10, 15\}$ minutes, respectively. Additionally, Eq. (5.3), configured with the settings from Section 5.3.1, has been used to delimit the maximum number of messages allowed for each recording interval. The results can be summarized as follows:
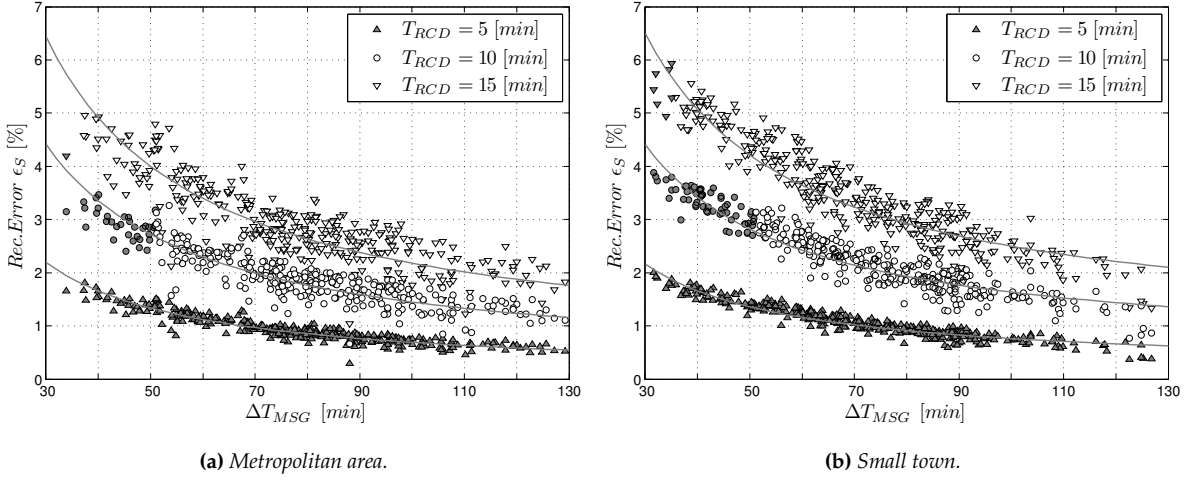
(a) *Metropolitan area.*   **(b)** *Small town.*

Figure 5.6. *Reconstruction accuracy vs. activity, for different recording intervals.*

- T_{RCD}=15$mins$: Eq. (5.4) is satisfied for $T_{MSG} \approx 34mins$. This means these settings guarantee that almost all of the analyzed sensors will arrive at the end of their planned life, i.e., the battery life can be guaranteed for the whole system at this sampling rate. Obviously, this set has the highest reconstruction error (inverted triangles).

- T_{RCD}=10$mins$: The limit value given by Eq. (5.4) for this case is $T_{MSG} \approx 51mins$. While most of the sensors are still within specifications (white circles), this value means that some of them will probably deplete their batteries before the end of their expected life (dark circles).

- T_{RCD}=5$mins$ (triangles): Operating with these settings means that none of the sensors will fulfill their expected life, independently of T_{MSG}. However, as expected, this set has the lowest reconstruction error.

Notably, the activity of the sensor clearly affects the reconstruction error of the occupancy (trends are represented with solid lines in the figure). This is something that is perhaps not obvious, but can be understood as follows: if a sensor is very active (because it is changing the state quite quickly) and the sampling rate is low, then the error introduced can be relatively high, as each change in the state can potentially induce some error. In other words, the sampling rate is too slow in comparison to the time the sensor remains in the same state.

5.3.3   Temporal Decimation: Adaptive Recording Interval

The recording interval settings can be adjusted from an alternative point of view. In this new approach, the sampling rate is no longer a system feature, but a control parameter for adjusting the power consumption of the system.

Specifically, as the time between records increases, less power is consumed. Then, by stretching the record interval it is possible to balance the extra power consumption associated to higher activity.

Section 5.2.3 provides the key for this adaptive technique. As stated previously, system activity can be considered constant for each specific sensor, in a statistical sense. Then, the recording interval T_{RCD} can be customized for each sensor so that the expected life can be guaranteed. Fig. 5.7 shows a scatter-plot representing the expected life of each sensor. The $T_{RCD} \leftrightarrow T_{MSG}$ mapping, given by Eq. (5.4), is estimated after a one-month learning stage, but the sensors are displayed at their actual measured T_{MSG}.

The main benefit of this customized sampling is that it homogenizes the expected life of sensors by balancing the power consumption, and thus it minimizes premature battery death or non-scheduled interventions.

Obviously, some fluctuations exist due to the uncertainty assumed when using a learning stage. However, with the exception of a few devices, the great majority of them can fulfill the expected life-time specification, at least according to everything that concerns the battery capacity
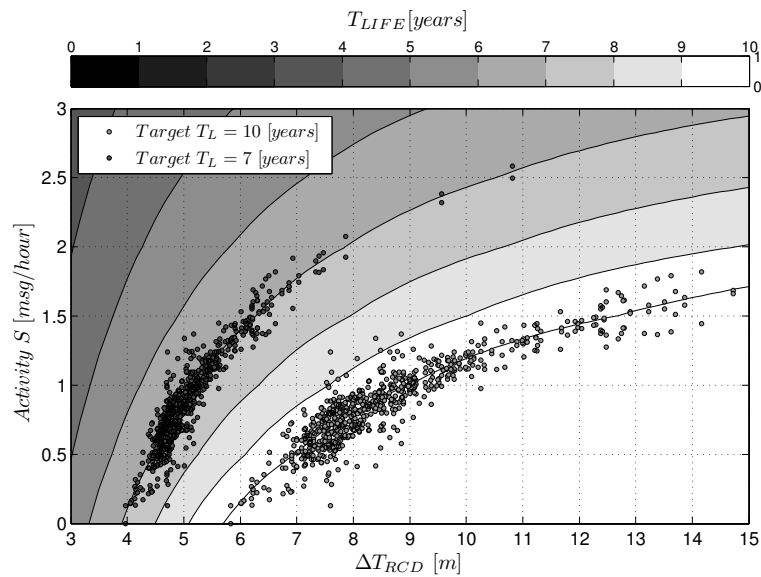

Figure 5.7. *Activity vs. adaptive record interval, for two different targets.*

The reconstruction error of each sensor can be evaluated for different recording intervals, which in turn are determined by the activity of the sensor (when energy is bounded). Fig. 5.8 shows the reconstruction error for each individual sensor, displayed as a function of the record interval. The results show devices running with adaptive sampling rate (triangles), superimposed to results expected for a fixed sampling rate: the gray circles indicate the error averaged for all sensors; the small black dots mark the value of each sensor (illustrating the variability in the system); and the black, solid line is the trend of the averaged errors.

Obviously, sensors with slower rates are expected to be less accurate, at least individually. Even more, when using adaptive sampling intervals that are above $T_{RCD} = 8mins$, the reconstruction error appears over the averaged trend of fixed settings (reflecting the fact that reconstruction error increases as the devices are more active, already observed in 5.6). A natural question arises as to how the error of these active devices affects to the reconstruction error of the whole system, and whether the negative effect of this small set could be compensated with the improved accuracy of less active devices that together form a larger set.
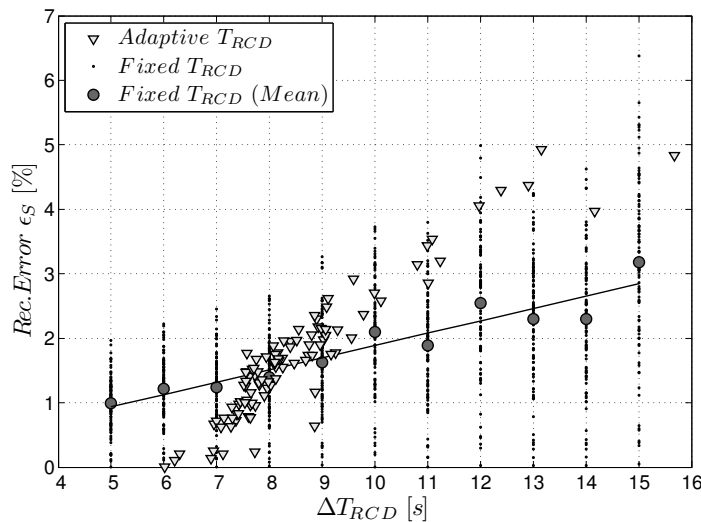

Figure 5.8. *Reconstruction Accuracy vs. Recording Interval (Sensor view).*

To answer this question, Fig. 5.9 shows the system reconstruction error ϵ_R vs. the record interval T_{RCD}. As expected, the system error is much lower than the individual contributions, as errors are statistically compensated.

The gray circles show the error when using a fixed sampling rate, and the dashed line indicates the trend when the (fixed) T_{RCD} is progressively longer. The black cross marks the reconstruction error of the adaptive sampling rate solution, placed at the averaged T_{RCD} of all sensors. Notably, the cross lies below the trend, which indicates that the better accuracy of most of the sensors in fact compensate for the worse accuracy of the small set of most active devices [1].

Moreover, the size of the circles in Fig. 5.9 is proportional to the number of sensors running with a fixed sampling rate whose expected life is shorter than required by specifications. The reconstruction error of the adaptive approach is approximately equal to the fixed one configured at $T_{RCD} = 8mins$; but with these settings, the latter causes 53% of the devices to have problems at the end of their service life

---

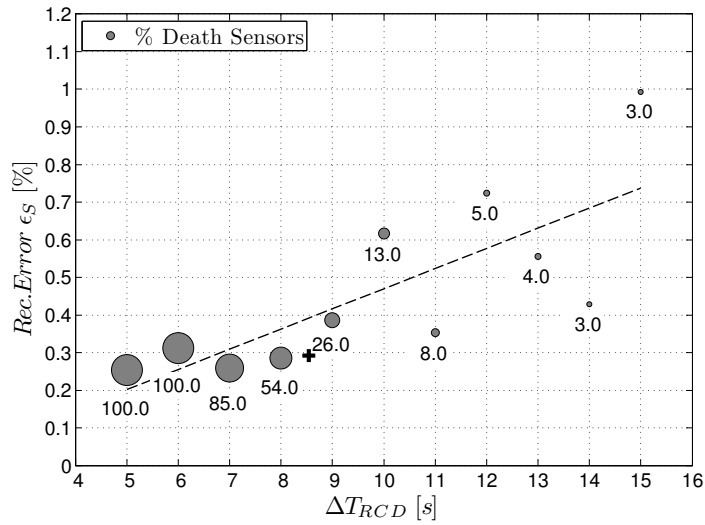[1] This asymmetry was seen in Fig. 5.5.

Figure 5.9. *Reconstruction Accuracy vs. Recording Interval (System). Estimate of the ratio potentially problematic devices due to premature death for $T_{RCD} = 10mins$.*

5.3.4 Reconstruction under Failure: Spatial Correlations

Independently of the followed approach, there is a certain risk of device failure before a scheduled intervention. Fig. 5.9 shows the ratio of potentially problematic devices when using a fixed record interval. However, even when using a customized record interval, some deviations from the predicted behavior may possibly occur in the learning stage, which itself may result in a premature death. Obviously, in addition to the battery life, the problem of physical device failure is always present. Therefore, this section aims to evaluate the accuracy of the system state reconstruction when the information is incomplete, whatever the reason for the failure.
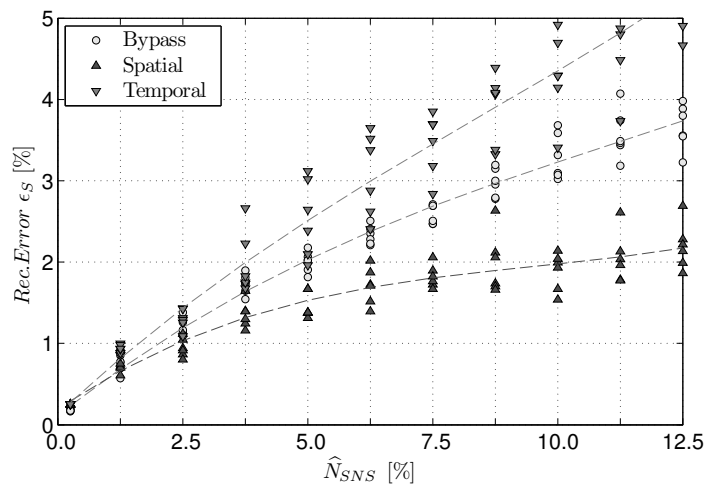

Figure 5.10. *Reconstruction error vs number of lost Sensors.*

Fig. 5.10 shows the results of different strategies for rebuilding the system occupancy state when the data associated to a set of devices is not available. Reconstruction error is plotted against the number of unavailable sensors from a total number of about 400. In this figure, each dot represents a period of 14 days. There are six dots for each coordinate value, corresponding to six 14-day periods over the three months. This particular arrangement is intended to verify that the temporal behavior is stable.

The figure shows three different sets of points, depending on the strategy taken:

- **Bypass Method**: (Shown in light gray.) Unavailable devices are simply omitted, and the state is built from the remaining sensors.

- **Temporal Self-Similarity Method**: (Shown in black.) Missing devices are modeled from their own past pattern, taking advantage of the sensor's self-similarity The obtained model is used to build the system state.

- **Spatial Similarity Method**: (Shown in dark gray.) Missing sensors are replaced by the nearest neighbor, making use of the spatial similarity.

Discussion

Following a real-world application as a case study, this chapter explores different alternatives for efficient energy management of wireless sensors, especially for those applications in which classical strategies such as data compression or local processing are not well suited.

This prospecting is carried out on two fronts. First, the monitoring service is understood as a service provided by multiple sensors with a common target: a system monitoring application. This view enables the introduction of system-level metrics, which in turn allows relaxing some specifications of individual sensors. Second, by introducing contextual information (i.e., the knowledge of the sensors temporal evolution and the surrounding state), the performance of individual sensors is improved.

As an application example, this chapter introduces an adaptive sampling rate method based on an historical evolution analysis of data. This method is designed to optimize energy usage by balancing the expected battery life among sensors. As discussed, this homogenization has a great impact on operational costs in that it minimizes non-scheduled interventions.

Nowadays, data is being acquired on an unprecedented scale, and the huge amount of information requires new methods of analysis and visualization. This chapter expands upon this new paradigm, in which spatio-temporal correlations are destined to play a central role.

Conclusions

The present study provides a systematic approach to defining energy efficiency policies for wireless devices based on an analysis of spatio-temporal correlations. This dissertation begins with and elaborates upon the hypothesis that exploiting spatio-temporal correlations leads to optimal strategies for ensuring highly efficient wireless sensor networks. To validate the formulated hypothesis, we have followed a deductive-inductive methodology by first introducing a generic energy profiling method. Second, we validate different spatio-temporal strategies through this method.

More specifically, the first result presented in this dissertation represents the basis of our thesis and consists of defining an accurate energy profiling model for wireless sensor devices. Interestingly, we found that such a model was lacking in the literature, even though it is essential in quantitatively validating the impact of different strategies for energy optimization. Hence, we have formalized a generic energy consumption methodology to profile the energy utilization of low-power, wireless embedded devices. The methodology aims to reduce technology adoption/integration risks by keeping in mind that energy consumption can be estimated precisely without any need for prototyping or building actual devices, as has been demonstrated by the presented work.

Throughout the first part of this thesis, we have established a generic framework based on energy flows for quantitatively determining the energy needs of specific WSN devices. More specifically, the proposed framework is built on three cornerstones: a) energy production characterization, b) a clear understanding of the device's energy usage; and c) an efficient temporary energy buffer management for coupling production with consumption patterns.

Importantly, the formulation of the model has revealed the existence of energy consumption cycles during WSN device operation, and it has shown that they can be a fundamental part of the energy consumption optimization strategies. The obtained results stress the importance of understanding the cycles of operation involved in embedded tasks, which inherently show repetitive behavior. As far as we know, ours is the first model that formalizes this idea and which has proven to be successful in different applications, as is demonstrated in the second part of this dissertation.

The second part of the thesis follows an inductive approach to demonstrating the applicability of spatio-temporal correlations under different scenarios, which we have found to be a valuable tool for defining energy efficiency policies. The main issue at hand is how to take advantage of the spatio-temporal correlations and quantitatively validate their impact using the proposed model from different angles.

The observed spatio-temporal correlations have been used to define energy efficiency policies, and their usefulness has been validated by the proposed model. Therefore, we can conclude that the main contribution of this work is that it demonstrates the following: spatio-temporal correlations play a central role in defining energy management policies, and thus they are essential in a generalist approach to addressing the problem of energy efficiency. This can be seen from the four different perspectives presented below.

From the specific applications perspective:

We have conducted this research by analyzing examples of several industrial applications. Each chapter deals with a real-world problem, which is analyzed and discussed in light of the energy efficiency model and the observed spatio-temporal correlations. Overall, the proposed methodology demonstrates its effectiveness in each of the different applications shown below.

- **Self-Sustainable Devices** (Chapter 3): Self-sufficient networks are destined to definitively cut down maintenance costs with devices that could operate unattended, thanks to the energy they extract from the environment. In this chapter, we have demonstrated that self-sustainability using off-the-shelf technologies is a reality today. Additionally, we have been able to quantify the conditions required for uninterrupted operation.

- **Data Compression** (Chapter 4): Compression is currently used for multiple purposes in embedded devices. Regardless of the suitability for energy savings, it can be very useful in reducing communication bandwidth and temporal storage space. In this chapter, we have exhaustively evaluated the performance of compression algorithms. Focusing on low-power applications, we have demonstrated that even though they may provide some energy savings, their usage should be carefully evaluated. Processing and transmission costs can be of the same order of magnitude and, depending on the specific algorithm and technology, the former may even dominate the latter.

- **Lean Sensing** (Chapter 5): Wireless sensors are only the hardware layer of today's complex, wide-area, monitoring systems. From a macro-scale point of view, information should be filtered, aggregated and correctly visualized to make it understandable. This step provides important energy and cost reduction opportunities, as we have demonstrated.

As an important result, we can conclude that the proposed generic framework proves to be valid for each of the proposed scenarios. In other words, this framework provides precise answers to concrete problems.

From the application domain perspective:

Taking a second point of view, this work has addressed low-power management strategies and demonstrated the usefulness of the proposed framework when considering the context. More specifically, the work has emphasized the importance of understanding contextual information obtained from spatio-temporal correlations. As a result we have proposed and evaluated different context-based strategies:

- **Event Detection** (Chapter 3): These applications are concerned mainly with detecting when certain events occur and generating alarms when they do. The distinctive characteristic is that all data processing is computed locally. Power reduction is achieved by eliminating radio transmissions.

- **Periodic Sampling and Reporting** (Chapter 4): These applications periodically sample and report all data acquired. Indeed, the imperfect reconstruction of the signal is often acceptable. Power reduction is achieved by reducing the amount of information to be transmitted by means of lossy data compression.

- **Periodic Sampling, Event Reporting** (Chapter 5): These applications periodically take samples, but they only generate a report if an event has been detected. Power can be reduced by relaxing the amount of data aggregated by the provided monitoring service.

As a general result, we can conclude that the different applications defined above can also be analyzed from a contextual perspective. The defined energy efficiency framework also applies here. More specifically for each of the contexts, we have proposed and validated an optimal strategy for efficient energy consumption by following the proposed framework.

From the quality of service perspective:

Throughout this dissertation, the main topics have been addressed with a common optimization objective. Indeed, we took special care to present applications as a tradeoff between the initial energy budget, the expected lifetime and the achievable quality of service:

- **Alarm-Based Applications** (Chapter 5): In this case, the monitoring service is failure detection, and the quality of service is measured by how often a periodic test can be performed.

- **Periodic Reporting Applications** (Chapter 4): For these applications, the service is the periodic information that is reported, and the quality tradeoff compromised is the accuracy of data reported.

- **Event Reporting Applications** (Chapter 3): For this application, the service is aggregated information visualization, and the quality is measured as the accuracy of these representations.

Again, this perspective allows us to conclude that the proposed framework is both valid and valuable in determining the tradeoff between service level agreements (quality of service) and the energy budget (which, in the end, determines the cost). This is very well aligned with industrial needs and service provisioning.

From the data perspective:

In a natural and progressive way, the spatio-temporal correlations have revealed hidden relationships within the application data, relationships that are not always clear or obvious. From a pure data analysis perspective, this has proven to be successful in leading to a better understanding of optimization strategies. Moreover, we have explored some visualization techniques in order to try to extract the patterns of a given data set. The observed correlations were essential in the following.

- **Energy Source** (Chapter 3): Energy harvesting is the most promising approach to addressing the limited lifetime of wireless devices. Patterns are inherently present in environmental energy sources. A clear understanding of these patterns proved to be essential in adjusting device power models according to energy availability.

- **Data Source** (Chapter 4): Limiting the amount of information to be communicated appears to be a natural choice for reducing radio energy, especially in those systems upon which this contribution carries significant weight. Self-correlations in the acquired data proved essential in the compressibility of physical information.

- **Information Source** (Chapter 5): Complex systems require aggregation and filtering techniques in order to be understood. Correlations emerge in the system analytics, as they are intrinsically related to aggregation. Thus, correlations can be used as a tool for interpreting a complex system while at the same time they are also useful in designing suitable policies based on this knowledge.

In summary, as we have discussed during the course of this dissertation, spatio-temporal correlations naturally appear in WSN data, and they can be successfully applied for optimizing the energy consumption of wireless devices. In general, we have proven that spatio-temporal correlations constitute a key element in optimally managing power consumption in low-power embedded devices.

· · ·

Mark Weiser envisioned an "ubiquitous computing" world [85]. Nowadays, the Internet of things is a reality, fulfilling the prophecy of Weiser's vision. In this word, a huge amount of information is ready for new methods of analysis and visualization. This work expands upon the knowledge of that can be gained within this new paradigm, in which we have demonstrated that spatio-temporal correlations are destined to play a central role.

List of Publications

The work presented in this thesis has appeared in the articles reported below.

Journal papers

[J1] I. Vilajosana, X. Vilajosana, J. Llosa, **B. Martinez**, M. Domingo-Prieto, A. Angles. "Bootstrapping Smart Cities Through A Self-Sustainable Model Based On Big Data Flows" *IEEE Communications Magazine*, Special Issue on Smart Cities. vol. 51, no. 6, June 2013.

[J2] D. Zordan, **B. Martinez**, I. Vilajosana and M. Rossi, "On the Performance of Lossy Compression Schemes for Energy Constrained Sensor Networking", *ACM Transactions on Sensor Networks.* vol. 11, no. 1, August 2014.

[J3] **B. Martinez**, X. Vilajosana, F. Charim, I. Vilajosana, K. Pister "When Scavengers meet Industrial Wireless" *IEEE Transactions on Industrial Electronics* October 2014.

Conference papers

[C1] M. Domingo-Prieto,**B. Martinez**, M. Montón, I. Vilajosana, X. Vilajosana and J. Arnedo-Moreno Balancing power consumption in IoT devices by using variable packet size 8th International Conference on Complex, Intelligent, and Software Intensive Systems CISIS 2014

Bibliography

[1] WirelessHART Specification 75: TDMA Data-Link Layer, 2008. HCF_SPEC-75.

[2] 802.15.4e-2012: IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer, 16 April 2012.

[3] National renewable energy laboratory, 2013. URL http://www.nrel.gov/rredc/.

[4] P. Abrahamsen. A Review of Gaussian Random Fields and Correlation Functions. Technical report, Norwegian Computing Center, Box 114, Blindern, N-0314 Oslo, Norway, 1997.

[5] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.*, 7(3):537–568, May 2009. ISSN 1570-8705.

[6] R. Arnott and J. Rowse. Modeling parking. *Journal of Urban Economics*, 45(1):97 – 124, 1999.

[7] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[8] K. C. Barr and K. Asanović. Energy-aware lossless data compression. *ACM Transactions on Computer Systems*, 24(3):250–291, Aug. 2006.

[9] M. Bazzaz, M. Salehi, and A. Ejlali. An accurate instruction-level energy estimation model and tool for embedded systems. *Instrumentation and Measurement, IEEE Transactions on*, 62(7):1927–1934, July 2013. ISSN 0018-9456.

[10] M. Belleville, H. Fanet, P. Fiorini, P. Nicole, M. Pelgrom, C. Piguet, R. Hahn, C. V. Hoof, R. Vullers, M. Tartagni, and E. Cantatore. Energy autonomous sensor systems: Towards a ubiquitous sensor technology. *Microelectronics Journal*, 41(11):740 – 745, 2010. {IEEE} International Workshop on Advances in Sensors and Interfaces 2009.

[11] T. Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression*. Prentice Hall, 1971.

[12] L. Bierl. MSP430 Family Mixed-Signal Microcontroller Application Reports. Technical report, Texas Instruments Incorporated, 2000.

[13] V. Boicea. Energy storage technologies: The past and the present. *Proceedings of the IEEE*, 102(11):1777–1794, Nov 2014.

[14] J. Brownjohn. Structural health monitoring of civil infrastructure. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 365(1851):589–622, 2007. ISSN 1364-503X.

[15] N. Bui and M. Rossi. Staying alive: System design for self-sufficient sensor networks. *ACM Transactions on Sensor Networks*, 2015.

[16] E. Candes and M. Wakin. An introduction to compressive sampling. *Signal Processing Magazine, IEEE*, 25 (2):21–30, March 2008. ISSN 1053-5888.

[17] H. Chen, J. Li, and P. Mohapatra. RACE: time series compression with rate adaptivity and error bound for sensor networks. In *IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Fort Lauderdale, FL, US, Oct. 2004.

[18] H. Chen, T. N. Cong, W. Yang, C. Tan, Y. Li, and Y. Ding. Progress in electrical energy storage system: A critical review. *Progress in Natural Science*, 19(3):291 – 312, 2009.

[19] T. Chen, Y. Yang, H. Zhang, H. Kim, and K. Horneman. Network energy saving technologies for green wireless access networks. *Wireless Communications, IEEE*, 18(5):30–38, October 2011. ISSN 1536-1284.

[20] Chipcon. SmartRF CC2420: 2.4 GHz IEEEE 802.15.4/ZigBee-ready RF Transceiver. Technical report, Texas Instruments Incorporated, 2007.

[21] R. Crepaldi, S. Friso, A. F. Harris III, M. Mastrogiovanni, C. Petrioli, M. Rossi, A. Zanella, and M. Zorzi. The Design, Deployment, and Analysis of SignetLab: A Sensor Network Testbed and Interactive Management Tool. In *IEEE Tridentcom*, Orlando, FL, US, May 2007.

[22] C. Dance. Lean smart parking. *The Parking Professional*, 30(6):26–29, june 2014.

[23] R. B. Davies and D. S. Harte. Tests for Hurst effect. *Biometrika*, 74(1):95–101, 1987.

[24] R. D'hulst, T. Sterken, R. Puers, G. Deconinck, and J. Driesen. Power processing circuits for piezoelectric vibration-based energy harvesters. *Industrial Electronics, IEEE Transactions on*, 57(12):4170–4177, 2010. ISSN 0278-0046.

[25] T. Dittrich, C. Menachem, H. Yamin, and L. Adams. Lithium batteries for wireless sensor networks, 2008.

[26] S. Dodson. The Internet of Things. *The Guardian*, oct 2003.

[27] L. Doherty, W. Lindsay, and J. Simon. Channel-specific wireless sensor network path data. In *ICCCN*, pages 89–94, 2007.

[28] D. Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, April 2006. ISSN 0018-9448.

[29] D. Donoho, M. Vetterli, R. DeVore, and I. Daubechies. Data compression and harmonic analysis. *Information Theory, IEEE Transactions on*, 44(6):2435–2476, 1998.

[30] K. D. Donohue. Wav data files for class examples and laboratory experiments. `http://www.engr.uky.edu/~donohue/ee422/Data/DataEE422.htm`, 2013.

[31] D. Estrin, D. Culler, K. Pister, and G. Sukhatme. Connecting the physical world with pervasive networks. *Pervasive Computing, IEEE*, 1(1):59–69, Jan 2002. ISSN 1536-1268.

[32] S. Evenepoel, J. VanOoteghem, S. Verbrugge, D. Colle, and M. Pickavet. On-street smart parking networks at a fraction of their cost: performance analysis of a sampling approach. *Transactions on Emerging Telecommunications Technologies*, 25(1):136–149, 2014. ISSN 2161-3915.

[33] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi. In-network aggregation techniques for wireless sensor networks: a survey. *Wireless Communications, IEEE*, 14(2):70–87, April 2007.

[34] D. Gunduz, K. Stamatiou, N. Michelusi, and M. Zorzi. Designing intelligent energy harvesting communication systems. *Communications Magazine, IEEE*, 52(1):210–216, January 2014.

[35] ISA. ISA-100.11a-2011: Wireless Systems for Industrial Automation: Process Control and Related Applications, May 2011.

[36] X. Jiang, J. Polastre, and D. Culler. Perpetual environmentally powered sensor networks. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 463–468, April 2005.

[37] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 6(4):32, 2007.

[38] A. Khaligh, P. Zeng, and C. Zheng. Kinetic energy harvesting using piezoelectric and electromagnetic technologies: State of the art. *Industrial Electronics, IEEE Transactions on*, 57(3):850–860, March 2010.

[39] S. Kim, R. Vyas, J. Bito, K. Niotaki, A. Collado, A. Georgiadis, and M. Tentzeris. Ambient rf energy-harvesting technologies for self-sustainable standalone wireless sensor platforms. *Proceedings of the IEEE*, 102(11):1649–1666, Nov 2014.

[40] A. N. Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers. In *Dokl. Akad. Nauk SSSR*, volume 30, pages 299–303, 1941.

[41] V. Konstantakos, A. Chatzigeorgiou, S. Nikolaidis, and T. Laopoulos. Energy consumption estimation in embedded systems. *Instrumentation and Measurement, IEEE Transactions on*, 57(4):797–804, April 2008. ISSN 0018-9456.

[42] Y. Liang. Efficient temporal compression in wireless sensor networks. In *International Conference on Local Computer Networks (LCN)*, Bonn, Germany, oct. 2011.

[43] C. Liu, K. Wu, and J. Pei. An energy-efficient data collection framework for wireless sensor networks by exploiting spatio-temporal correlation. *IEEE Transactions on Parallel and Distributed Systems*, 18(7):1010–1023, jul 2007.

[44] B. Lu and V. Gungor. Online and remote motor energy monitoring and fault diagnostics using wireless sensor networks. *Industrial Electronics, IEEE Transactions on*, 56(11):4651–4659, 2009. ISSN 0278-0046.

[45] J.-L. Lu, F. Valois, and M. Dohler. Optimized Data Aggregation in WSNs Using Adaptive ARMA. In *International Conference on Sensor Technologies and Applications (SENSORCOMM)*, Venice, Italy, jul 2010.

[46] G. Mao, B. D. O. Anderson, and B. Fidan. Online Calibration of Path Loss Exponent in Wireless Sensor Networks. In *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, San Francisco, CA, US, Nov. 2006.

[47] F. Marcelloni and M. Vecchio. Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization. *Elsevier Information Sciences*, 180(10):1924–1941, May 2010.

[48] G. Marrocco. Pervasive electromagnetics: sensing paradigms by passive rfid technology. *Wireless Communications, IEEE*, 17(6):10–17, December 2010. ISSN 1536-1284.

[49] A. Mehta and K. Pister. WARPWING: A Complete Open-Source Control Platform for Miniature Robots. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2010.

[50] A. Muszynska. Vibrational diagnostics of rotating machinery malfunctions. *International Journal of Rotating Machinery*, 1(3-4):237–266, 1995.

[51] A. Nasiri, S. Zabalawi, and G. Mandic. Indoor power harvesting using photovoltaic cells for low-power applications. *Industrial Electronics, IEEE Transactions on*, 56(11):4502–4509, 2009. ISSN 0278-0046.

[52] S. Nikolaidis, N. Kavvadias, T. Laopoulos, L. Bisdounis, and S. Blionas. Instruction level energy modeling for pipelined processors. *J. Embedded Comput.*, 1(3):317–324, Aug 2005. ISSN 1740-4460.

[53] J. Paradiso and T. Starner. Energy scavenging for mobile and wireless electronics. *Pervasive Computing, IEEE*, 4(1):18–27, Jan 2005. ISSN 1536-1268.

[54] P. Peebles. *The large scale structure of the universe*. Princeton University Press, Princeton, NY, USA, 1980.

[55] A. Pernia, S. Menendez, M. Prieto, J. Martinez, F. Nuno, I. Villar, and V. Ruiz. Power supply based on carbon ultracapacitors for remote supervision systems. *Industrial Electronics, IEEE Transactions on*, 57(9):3139–3147, Sept 2010.

[56] N. D. Pham, T. D. Le, and H. Choo. Enhance exploring temporal correlation for data collection in WSNs. In *IEEE International Conference on Research, Innovation and Vision for the Future (RIVF)*, Ho Chi Minh City, Vietnam, jul 2008.

[57] G. M. Phillips. *Interpolation and Approximation by Polynomials*. Springer, 2003.

[58] G. J. Pottie and W. J. Kaiser. Wireless integrated network sensors. *Commun. ACM*, 43(5):51–58, May 2000. ISSN 0001-0782.

[59] M. Presser. *Smart Cities at the Forefront of the Future Internet*, volume 6656 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2011.

[60] G. Quer, R. Masiero, G. Pillonetto, M. Rossi, and M. Zorzi. Sensing, Compression and Recovery for WSNs: Sparse Signal Modeling and Monitoring Framework. *IEEE Transactions on Wireless Communications*, 11(10):3447–3461, Oct. 2012.

[61] K. R. Rao and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Academic Press Professional, Inc., San Diego, CA, US, 1990.

[62] F. Rimmer. Transmission line operations and maintenance. *Electric Energy Magazine*, April 2005.

[63] S. Roundy, D. Steingart, L. Frechette, P. Wright, and J. Rabaey. *Power Sources for Wireless Sensor Networks*, volume 2920 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004. ISBN 978-3-540-20825-9.

[64] T. Schoellhammer, B. Greenstein, E. Osterweil, M. Wimbrow, and D. Estrin. Lightweight temporal compression of microclimate datasets. In *IEEE International Conference on Local Computer Networks (LCN)*, Tampa, FL, US, Nov. 2004.

[65] Semtech. LoRa Product Family. `http://www.semtech.com/wireless-rf/lora.html`, 2014.

[66] SigFox. About SigFox. `http://www.sigfox.com/en/about`, 2014.

[67] F. Simjee and P. Chou. Efficient charging of supercapacitors for extended lifetime of wireless sensor nodes. *Power Electronics, IEEE Transactions on*, 23(3):1526–1536, May 2008.

[68] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, and K. Araki. Practical data compression in wireless sensor networks: A survey. *Elsevier Journal of Network and Computer Applications*, 35(1):37–59, 2012.

[69] Y. Tan and S. Panda. Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes. *Industrial Electronics, IEEE Transactions on*, 58(9):4424–4435, 2011. ISSN 0278-0046.

[70] W. R. Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, 46: 234–240, 1970.

[71] R. Torah, P. Glynne-Jones, M. Tudor, T. O'Donnell, S. Roy, and S. Beeby. Self-powered autonomous wireless sensor node using vibration energy harvesting. *Measurement Science and Technology*, 19(12):125202, 2008.

[72] K. Tutuncuoglu and A. Yener. Optimum transmission policies for battery limited energy harvesting nodes. *Wireless Communications, IEEE Transactions on*, 11(3):1180–1189, March 2012. ISSN 1536-1276.

[73] A. Van Der Byl, R. Neilson, and R. Wilkinson. An evaluation of compression techniques for Wireless Sensor Networks. In *AFRICON*, Nairobi, Kenya, sep 2009.

[74] I. Vilajosana, G. Khazaradze, E. Suriñach, E. Lied, and K. Kristensen. Snow avalanche speed determination using seismic methods. *Elsevier Cold Regions Science and Technology*, 49(1):2–10, july 2007.

[75] X. Vilajosana and K. Pister. Ietf internet draft. 6tisch wg document. minimal 6tisch configuration. draft-ietf-6tisch-minimal, 2014. URL `https://datatracker.ietf.org/doc/draft-ietf-6tisch-minimal/`.

[76] X. Vilajosana, Q. Wang, F. Chraim, T. Watteyne, T. Chang, and K. Pister. A realistic energy consumption model for tsch networks. *Sensors Journal, IEEE*, 14(2):482–489, Feb 2014.

[77] L. D. Volker Buscher and D. Hill. Smart cities: Transforming the 21st century city via the creative use of technology, 2010.

[78] R. Vullers, R. van Schaijk, I. Doms, C. V. Hoof, and R. Mertens. Micropower energy harvesting. *Solid-State Electronics*, 53(7):684 – 693, 2009.

[79] R. Vullers, R. Schaijk, H. Visser, J. Penders, and C. Hoof. Energy harvesting for autonomous wireless sensor networks. *Solid-State Circuits Magazine, IEEE*, 2(2):29–38, Spring 2010. ISSN 1943-0582.

[80] G. K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38 (1):xviii –xxxiv, Feb. 1992.

[81] Q. Wang, M. Hempstead, and W. Yang. A realistic power consumption model for wireless sensor network devices. In *Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on*, volume 1, pages 286–295, 2006.

[82] A. Waterbury and P. K. Wright. Vibration energy harvesting to power condition monitoring sensors for industrial and manufacturing equipment. *Proc. of the Institution of Mech. Engineers, Part C: Journal of Mechanical Engineering Science*, 227(6):1187–1202, 2013.

[83] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. Glaser, and K. Pister. Openwsn a standards-based low-power wireless development environment. *Transactions on Emerging Telecommunications Technologies*, 23(5):480–493, 2012.

[84] W. Webb. Weightless: The technology to finally realise the m2m vision. *Int. J. Interdiscip. Telecommun. Netw.*, 4(2):30–37, apr 2012. ISSN 1941-8663.

[85] M. Weiser. The computer for the 21st century. *Scientific american*, 265(3):94–104, 1991.

[86] T. Yin, H. Lam, H. Chow, and H. Zhu. Dynamic reduction-based structural damage detection of transmission tower utilizing ambient vibration data. *Engineering Structures*, 31(9):2009 – 2019, 2009.

[87] D. Zordan, G. Quer, M. Zorzi, and M. Rossi. Modeling and Generation of Space-Time Correlated Signals for Sensor Network Fields. In *IEEE Global Telecommunications Conference (GLOBECOM)*, Houston, TX, US, Dec. 2011.

[88] D. Zordan, B. Martinez, I. Vilajosana, and M. Rossi. On the performance of lossy compression schemes for energy constrained sensor networking. *ACM Trans. Sen. Netw.*, 11(1):15:1–15:34, Aug 2014.

Epilogue

'THAT Hooker telescope of yours is quite spectacular. I can imagine how much fun young Humason and you have with it. Your tour was full of such lovely details too, just like your papers.'

Edwin Hubble laughed. 'Come, come, Albert. This flattery doesn't suit you. It would be an honour, if you would tell me what you really think of my work.' He emphasized the word 'really'.

It was a pleasant afternoon in January in 1931. Mount Wilson Observatory was playing host to a group of special guests. One of these guests, the most special one in the eyes of Hubble, was his respected colleague, the renowned scientist, Albert Einstein. The tour was done and the two scientists had some time alone. They had exchanged letters in the last decade or so but Hubble felt that many things had been left unsaid in the written word.

'Well, you must know this already. But I did tell de Sitter long ago that the possibilities of an "expanding" universe seemed quite senseless to me. I know you have proof but I still find it hard to make my peace with it,' said Einstein.

Hubble was quiet. He put his coffee mug on the desk and walked to the windows of his office.

'Look, Albert. We are over five thousand feet high. It's such a nice day, that even the Santa Catalina Island is visible. It was a day like this, a beautiful Californian day with each star refusing to compromise on showcasing its brilliance, when I first discovered that the redshift of distant galaxies increased as a linear function of their distance. Of course I couldn't believe it either. Our magnificent universe was expanding? It was a mad thought much bigger than any of us. How was I going to change people's minds about what they thought they already knew? How was I going to tell people that the universe wasn't as static as they had imagined? But despite these doubts, I was thrilled. It boggled my mind and ignited my imagination. I can tell you now that I am convinced, our magnificent universe is expanding.'

'Einstein took a sip of his coffee. We have to be a bit mad as scientists. I don't blame you.'

'Some would say that my madness was consistent with yours,' said Hubble. 'Well, all of your madness except the bit that came up with the *lambda*. Many people didn't believe my discovery because it didn't fit with your equations of relativity. The existence of your cosmological constant, lambda suggested the universe was static.'

'If you recall, that crazy Belgian Catholic priest, Lemaître had proposed the same concept of the expanding universe. And his work actually fit quite well into my theories of relativity.'

'Why then, were you so critical of a non static universe? You must have been aware that the only way people would believe Lemaître's and my theories was if you got rid of the Cosmological Constant from your field equations of relativity.'

'I must admit that your proof of the redshift of distant nebulae has smashed my old construction like a hammer blow,' replied Einstein. Hubble's eyebrows shot up with expectation. Einstein went on, 'When I first came up with the general theory of relativity, I was amazed, if I may say so myself, by their simplicity and beauty. But assuming the universe was not static; my equations were telling me that the universe would collapse into itself or keep expanding. I will admit that the constant lambda was necessary for the purpose of making possible a quasi-static distribution of matter.'

Hubble smiled his first smile that afternoon. 'Am I hearing what I think I'm hearing?'

'You will hear what you wish to hear, my dear Edwin. I have been thinking for a while now that the constant has to go. Everything points in the direction of an expanding universe. I will just have to admit to the world that I made an error in my calculations. I won't deny that I'm a little pleased to have my symmetric equations of relativity back. You have my congratulations and my thanks for your persistence. I still need to think about eliminating the constant completely but I feel I'm moving closer in that direction.' He looked at his pocket watch. 'And now, i's time to go. Thank you for a delightful time. I'm sure we will see each other again.'

The two men shook hands and walked out of the door together. The rest of the group was at the end of the corridor. Just as they arrived towards them, Hubble said in a quiet tone to Einstein, 'I know what you always say, Albert, that anyone who has never made a mistake has never tried anything new. Your instinct is other worldly. As a universe, we might be expanding, and you will be right to

dismiss your constant so we can share this discovery with the world and they will believe us... but... '

'But, you wouldn't be surprised if the cosmological constant came back again one day?' Einstein laughed. 'You certainly have a charming sense of humour, Edwin. But who is to say? You never know what can happen in this world, or where we are going.'
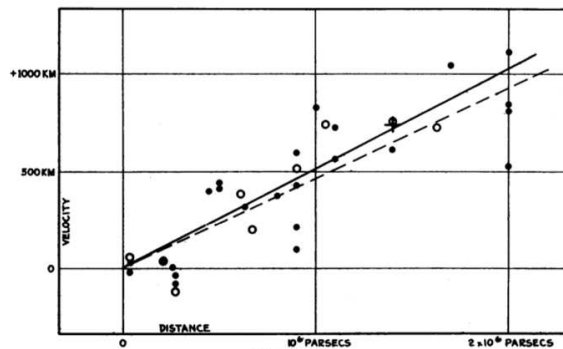
…

FIGURE 1

Velocity-Distance Relation among Extra-Galactic Nebulae.

Hubble's theories supported the scientist Lemaître in his claim that the universe is the 'ashes and smoke of bright but very rapid fireworks;' a spark that ignited the 'Big Bang Theory'.

In 1998, many years after Einstein's visit to Hubble at Mount Wilson and his dismissal of the constant 'lambda' from his equations of relativity, two independent teams of astronomers, the Supernova Cosmology Project and the High-z Supernova Search Team, announced a remarkable discovery — the rate at which the universe is expanding is accelerating. This discovery suggested the existence of 'dark energy' in the universe, something that could only be proven by re-introducing Einstein's lambda back into his equations. Some might say that the cosmological constant had come back.

"Conversations About A Constant" — Sneha Nagesh